



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA

INGENIERÍA CIVIL – HIDRÁULICA

PREDICCIÓN DE ESCURRIMIENTO MEDIO DIARIO MEDIANTE REDES
NEURONALES ARTIFICIALES EN LA PRESA LA ANGOSTURA, CHIAPAS.

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA

PRESENTA:

MACKENDY CERAGENE

TUTORES PRINCIPALES

DRA. MARITZA LILIANA ARGANIS JUÁREZ
DR. RAMÓN DOMÍNGUEZ MORA

INSTITUTO DE INGENIERÍA
INSTITUTO DE INGENIERÍA

CIUDAD UNIVERSITARIA, CD. MX., JUNIO 2023



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

JURADO ASIGNADO:

Presidente: Dr. Fuentes Mariles Oscar Arturo
Secretario: Dr. Domínguez Mora Ramón
1 er. Vocal: Dra. Arganis Juárez Maritza Liliana
2 do. Vocal: M. I. Franco Víctor
3 er. Vocal: Dr. Pozos Estrada Óscar

Instituto de ingeniería, UNAM, Ciudad Universitaria, CD. MX.

TUTORES DE TESIS:

Dra. Arganis Juárez Maritza Liliana

Dr. Domínguez Mora Ramón



FIRMA



FIRMA

NOMBRE DEL TRABAJO

TESIS MCKENDY UNAM_2.pdf

AUTOR

MCKENDY CERAGENE

RECUENTO DE PALABRAS

26018 Words

RECUENTO DE CARACTERES

145675 Characters

RECUENTO DE PÁGINAS

110 Pages

TAMAÑO DEL ARCHIVO

9.1MB

FECHA DE ENTREGA

Jun 1, 2023 6:53 PM CST

FECHA DEL INFORME

Jun 1, 2023 6:55 PM CST**● 6% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos

- 4% Base de datos de Internet
- Base de datos de Crossref
- 3% Base de datos de trabajos entregados
- 2% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

● Excluir del Reporte de Similitud

- Material bibliográfico
- Material citado
- Bloques de texto excluidos manualmente
- Material citado
- Coincidencia baja (menos de 15 palabras)

EL PRESENTE TRABAJO DE INVESTIGACIÓN DE TESIS SE LLEVÓ A CABO EN EL INSTITUTO DE INGENIERÍA DE LA UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO, CIUDAD DE MÉXICO, MÉXICO.

DEDICATORIA

A todas aquellas personas que han perdido un ser de su familia durante la pandemia COVID-19 y a los damnificados de eventos hidrometeorológicos en el mundo.

AGRADECIMIENTOS

A toda mi familia y mi novia Lucy, por creer en mí, apoyarme en todas las formas posibles y por haberme demostrado un profundo amor incondicional y hacerme sentir que no estoy solo.

A mis asesores la Dra. Maritza Liliana Arganis Juárez y el Dr. Ramón Domínguez Mora, por su admirable dedicación, quiénes además de sus conocimientos y aporte como directores, me apoyaron y me ayudaron de una manera extraordinaria, sin ellos la realización de este trabajo de tesis en tiempo y forma no habría sido posible.

A mis profesores por compartir sus conocimientos conmigo y miembros de mi comité tutorial por su disponibilidad y asesoramientos durante toda la maestría.

A mis amigos que me brindaron su apoyo, consejos, amistad y me acompañaron durante toda esta etapa de mi vida.

Al Instituto de Ingeniería de la UNAM por el apoyo económico otorgado con el que fue posible llevar a cabo el estudio de la maestría en ingeniería civil y el término de este trabajo de investigación de tesis.

CONTENIDO

| | |
|---|-----------|
| RESUMEN..... | 1 |
| ABSTRACT..... | 2 |
| INTRODUCCIÓN..... | 3 |
| CAPÍTULO 1:..... | 5 |
| 1. GENERALIDADES DE REDES NEURONALES ARTIFICIALES | 5 |
| 1.1. ANTECEDENTES | 5 |
| 1.2. PLANTEAMIENTO DEL PROBLEMA | 8 |
| 1.3. JUSTIFICACIÓN | 9 |
| 1.4. OBJETIVO GENERAL | 10 |
| 1.5. OBJETIVOS ESPECÍFICOS | 10 |
| 1.6. HIPÓTESIS | 11 |
| 1.7. ORIGEN DE LAS REDES NEURONALES | 11 |
| CAPÍTULO 2:..... | 13 |
| 2. MARCO TEÓRICO | 13 |
| 2.1. CARACTERÍSTICAS DE LAS REDES NEURONALES ARTIFICIALES | 13 |
| 2.2. ESTRUCTURA BÁSICA DE UNA RED NEURONAL..... | 13 |
| 2.2.1. Analogía con el Cerebro..... | 13 |
| 2.2.2. Redes Neuronales Artificiales | 16 |
| 2.3. DEFINICIONES DE UNA RED NEURONAL..... | 17 |
| 2.4. VENTAJAS QUE OFRECEN LAS REDES NEURONALES..... | 17 |
| 2.5. APLICACIONES DE LAS REDES NEURONALES | 20 |
| 2.6. ARQUITECTURA DE LAS REDES NEURONALES | 22 |
| 2.6.1. Clasificación por tipo de capas..... | 22 |
| 2.6.2. Clasificación por número de capas | 24 |

| | | |
|----------|--|----|
| 2.7. | MODELO DE UNA RED NEURONAL..... | 25 |
| 2.7.1. | Entrada y salida..... | 26 |
| 2.7.2. | Pesos | 26 |
| 2.7.3. | Clasificador lineal o regla de propagación..... | 27 |
| 2.7.4. | Función de activación..... | 27 |
| 2.7.4.1. | Características de función de activación..... | 28 |
| 2.7.4.2. | Tipos de función de activación | 29 |
| 2.8. | TOPOLOGÍA DE REDES NEURONALES | 33 |
| 2.9. | FASE DE ENTRENAMIENTO O APRENDIZAJE DE LA RED NEURONAL . | 35 |
| 2.9.1. | Ajuste de Hiperparámetros..... | 37 |
| 2.9.2. | Tipo de aprendizaje..... | 39 |
| 2.9.2.1. | Aprendizaje supervisado..... | 39 |
| 2.9.2.2. | Aprendizaje no supervisado..... | 40 |
| 2.9.2.3. | Aprendizaje por refuerzo..... | 41 |
| 2.10. | VALIDACIÓN O FASE DE PRUEBA U OPERACIÓN DE LA RED..... | 41 |
| 2.10.1. | Índices de evaluación del pronóstico..... | 42 |
| 2.11. | ALGORITMO DE RETROPROPAGACIÓN | 44 |
| 2.12. | MODELOS ESTOCÁSTICOS | 45 |
| 2.12.1. | Modelo AR(p) | 45 |
| 2.12.2. | Modelo ARMA (p, q) | 52 |
| 2.12.3. | Modelos de pronóstico | 56 |
| 2.12.4. | Modelo ARIMA (p, d, q)..... | 58 |
| 2.13. | CARACTERÍSTICAS ESTADÍSTICAS | 62 |
| 2.13.1. | Medidas de tendencia central..... | 62 |
| 2.13.2. | Medidas de dispersión | 63 |

| | | |
|------------------------|---|-----------|
| 2.13.3. | Medidas de posición..... | 64 |
| 2.13.4. | Medidas de distribución..... | 65 |
| CAPÍTULO 3: | | 66 |
| 3. | METODOLOGÍA..... | 66 |
| 3.1. | MEDIO FÍSICO..... | 66 |
| 3.1.1. | Zona de estudio..... | 66 |
| 3.1.2. | Información disponible..... | 68 |
| 3.1.3. | Procesamiento de los datos..... | 71 |
| 3.2. | MODELOS ESTOCÁSTICOS..... | 71 |
| 3.2.1. | Prueba Dickey-Fuller Aumentada (ADF)..... | 72 |
| 3.2.2. | Separación de los datos..... | 74 |
| 3.2.3. | Selección del mejor modelo estocástico..... | 75 |
| 3.3. | MODELOS DE REDES NEURONALES ARTIFICIALES..... | 83 |
| CAPÍTULO 4: | | 90 |
| 4. | DISCUSIONES DE RESULTADOS..... | 90 |
| 4.1. | PRONÓSTICO CON EL MEJOR MODELO ESTOCÁSTICO..... | 90 |
| 4.2. | PRONÓSTICO CON EL MEJOR MODELO DE RED NEURONAL..... | 91 |
| 4.3. | COMPARACIÓN ENTRE LOS DOS MEJORES MODELOS DE ESTOCÁSTICO Y RED NEURONAL ARTIFICIAL..... | 93 |
| CONCLUSIONES | | 95 |
| RECOMENDACIONES | | 97 |
| REFERENCIAS | | 98 |

ÍNDICE DE FIGURAS

| | |
|--|-----------|
| <i>Figura 2.1. Estructura general de una neurona biológica.....</i> | <i>15</i> |
| <i>Figura 2.2. Diagrama de una neurona artificial.</i> | <i>16</i> |
| <i>Figura 2.3. Arquitectura unidireccional de una red neuronal (Meneses, 2017).</i> | <i>23</i> |
| <i>Figura 2.4. Arquitectura de redes monocapa y multicapa.....</i> | <i>25</i> |
| <i>Figura 2.5. Función de activación de una red neuronal.</i> | <i>28</i> |
| <i>Figura 2.6. Ejemplo de cálculo del valor de función de activación de una neurona.....</i> | <i>33</i> |
| <i>Figura 2.7. Algunas topologías de RNA.</i> | <i>35</i> |
| <i>Figura 3.1. Principales presas en México (Conagua, 2018).....</i> | <i>67</i> |
| <i>Figura 3.2. Ubicación de la zona de estudio (Elaboración propia).....</i> | <i>68</i> |
| <i>Figura 3.3. Información disponible en la zona de estudio.</i> | <i>69</i> |
| <i>Figura 3.4. Estaciones hidrométricas seleccionadas y sus respectivas subcuencas.</i> | <i>70</i> |
| <i>Figura 3.5. Máximos anuales de estaciones hidrométricas.</i> | <i>72</i> |
| <i>Figura 3.6. Prueba Dickey-Fuller Aumentada con datos originales y con retraso 365.....</i> | <i>73</i> |
| <i>Figura 3.7. Prueba Dickey-Fuller Aumentada con datos $d=1$ y con retraso 365.....</i> | <i>74</i> |
| <i>Figura 3.8. Separación de los datos de las estaciones hidrométricas.....</i> | <i>75</i> |
| <i>Figura 3.9. Gráfica PACF con retraso 90.</i> | <i>76</i> |
| <i>Figura 3.10. Pronósticos de la fase de entrenamiento de $AR(p)$.....</i> | <i>78</i> |
| <i>Figura 3.11. Gráfica ACF con retraso 90.</i> | <i>78</i> |
| <i>Figura 3.12. Pronósticos de la fase de entrenamiento de $ARMA(p, q)$.....</i> | <i>80</i> |
| <i>Figura 3.13. Pronósticos de la fase de entrenamiento de $ARIMA(p, d, q)$.....</i> | <i>82</i> |
| <i>Figura 3.14. Función de coste de cada modelo de red neuronal.....</i> | <i>87</i> |
| <i>Figura 3.15. Pronósticos de la fase de entrenamiento de modelos de red neuronal LSTM.....</i> | <i>89</i> |
| <i>Figura 4.1. Pronósticos de la fase de validación de modelos estocásticos.</i> | <i>91</i> |
| <i>Figura 4.2. Pronósticos de la fase de validación de modelos RNA de LSTM.....</i> | <i>93</i> |
| <i>Figura 4.3. Ecuaciones de regresión lineal de una recta $y = ax + b$ de cada modelo.....</i> | <i>94</i> |

ÍNDICE DE TABLAS

| | |
|---|----|
| <i>Tabla 3.1. Presas del sistema Grijalva con capacidad instalada y generación media anual (INEGI, 2014).</i> | 67 |
| <i>Tabla 3.2. Información hidrométrica utilizada.</i> | 71 |
| <i>Tabla 3.3. Características estadísticas de las estaciones hidrométricas.</i> | 71 |
| <i>Tabla 3.4. Parámetros de modelos autorregresivos.</i> | 76 |
| <i>Tabla 3.5. Índices de comparación de modelos AR(p).</i> | 77 |
| <i>Tabla 3.6. Parámetros de modelos autorregresivos de medios móviles.</i> | 79 |
| <i>Tabla 3.7. Índices de comparación de modelos ARMA(p, q).</i> | 79 |
| <i>Tabla 3.8. Parámetros de modelos auto regresivos integrados de medios móviles.</i> | 81 |
| <i>Tabla 3.9. Índices de comparación de ARIMA(p, d, q).</i> | 81 |
| <i>Tabla 3.10. Índices de pronósticos del entrenamiento de modelos estocásticos.</i> | 83 |
| <i>Tabla 3.11. Valores de búsqueda del ajuste de hiperparámetros.</i> | 86 |
| <i>Tabla 3.12. Ajuste de hiperparámetros de las mejores combinaciones.</i> | 87 |
| <i>Tabla 3.13. Índices de pronósticos de escenarios de la red neuronal.</i> | 88 |
| <i>Tabla 4.1. Índices de pronósticos de validación de los modelos estocásticos.</i> | 90 |
| <i>Tabla 4.2. Índices de pronósticos de validación de los modelos de RNA de LSTM.</i> | 92 |
| <i>Tabla 4.3. Índices de comparación de los dos mejores modelos de pronóstico.</i> | 93 |

RESUMEN

La inteligencia artificial (IA) puede utilizarse para predecir variables climatológicas utilizando modelos de aprendizaje automático. Estos modelos pueden analizar grandes cantidades de datos históricos, como mediciones atmosféricas, patrones climáticos pasados y otras variables relevantes, para identificar tendencias y patrones que ayuden a predecir el clima futuro. En las últimas dos décadas (a partir 2002) y con el nacimiento de disciplina de la Inteligencia Artificial e Hidrología que es la *Hidroinformática*, que incluyen procedimientos como las Redes Neuronales Artificiales (RNA) que han ido en aumento en su número de artículos científicos publicados sobre aplicación en diferentes estudios de aprovechamientos, conservación de recursos hídricos y en particular sobre el tema de predicción de variables hidroclimatológicas.

En este contexto, el objetivo principal de este trabajo de tesis es pronosticar caudal [m^3/s] medio diario de entrada en la presa La Angostura, Chiapas. Primero, mediante una comparación de modelos de pronóstico Autorregresivo AR(p), Autorregresivo de Medias Móviles ARMA (p,q), Autorregresivo Integrado de Medias Móviles (p,d,q), conforme se aumenta el número de parámetros para obtener una mayor precisión del pronóstico y comparar el mejor modelo estocástico con el mejor modelo de redes neuronales artificiales de tipo recurrente. Se usaron datos de caudal medio diario de tres estaciones hidrométricas con periodo común de 1977 a 2007; para los primeros 29 años se utilizaron para el entrenamiento y los últimos dos años fueron para la validación.

En los modelos AR, ARMA y ARIMA se utilizaron $p = q = 1,2,3,4$ y $d = 1$ mientras se hizo un análisis de hiperparámetros para seleccionar 3 mejores modelos de red neuronal artificial recurrente LSTM. Para la evaluación y comparación de los modelos se utilizaron Coeficiente de Correlación (R), Raíz del Error Medio Cuadrático ($RMSE$), Error Medio Absoluto (MAE) y Error Medio Porcentual Absoluto ($MAPE$). Para el pronóstico de escurrimiento en la fase de entrenamiento y validación, el modelo ARIMA tuvo menor error que los modelos AR y ARMA. Finalmente, se eligió el modelo de red neuronal de LSTM como el mejor modelo para pronosticar escurrimiento medio diario en el embalse, es capaz de pronosticar el 98% de los escurrimientos medio diario de entrada en la presa del pasado 730 días.

ABSTRACT

Artificial intelligence (AI) can be used to predict weather variables using machine learning models. These models can analyze large amounts of historical data, such as atmospheric measurements, past weather patterns, and other relevant variables, to identify trends and patterns that help predict future climate. In the last two decades (from 2002) and with the birth of discipline of Artificial Intelligence and Hydrology that is Hydro informatics, which include procedures such as Artificial Neural Networks (RNA) that have been increasing in the number of articles scientific publications on application in different studies of exploitation, conservation of water resources and about prediction of hydroclimatological variables.

In this context, the main objective of this thesis work is to forecast the average daily inlet flow [m^3/s] at the La Angostura dam, Chiapas. First, by a comparison of autoregressive AR(p), autoregressive Moving Averages ARMA (p,q), Integrated autoregressive Moving Averages (p,d,q) forecast models while increasing the number of parameters to obtain higher forecast accuracy and compare the best stochastic model with the best model of recurrent artificial neural networks. Daily flow data from three hydrometric stations with a common period from 1977 to 2007 were used, for the first 29 years were used for training and the last two years were for validation.

In the AR, ARMA and ARIMA models, $p=q=1,2,3,4$ and $d=1$ was used while performing a hyper parameter analysis to select 3 best LSTM recurrent artificial neural network models. For the evaluation and comparison of the models, the Correlation Coefficient (R), Root Mean Squared Error ($RMSE$), Mean Absolute Error (MAE) and Mean Absolute Percentage Error ($MAPE$) were used. For the runoff forecast in the training and validation phase, the ARIMA model had less error than the AR and ARMA models. Finally, the LSTM neural network model was chosen as the best model to forecast mean daily runoff in the reservoir, it is capable of forecasting 98% of the mean daily runoff entering the dam for the past 730 days.

INTRODUCCIÓN

Las redes neuronales artificiales (RNA) se utilizan en la hidrología para abordar diferentes problemas y aplicaciones, tales como predicción de caudales, modelado de procesos hidrológicos, calibración de modelos hidrológicos, análisis de incertidumbre. Para predecir caudales en ríos o cuencas hidrográficas se utilizan datos históricos de caudales, datos climáticos y otras variables relevantes, se puede entrenar una RNA para aprender las relaciones y patrones entre estas variables y realizar predicciones de caudales futuros. Esto puede ser útil para la gestión del agua, la planificación de infraestructuras hidroeléctricas, el control de inundaciones y otras aplicaciones relacionadas con el agua.

Es importante destacar que las RNA no son la única herramienta utilizada en hidrología, y su aplicación debe complementarse con otros enfoques y considerar las características específicas del sistema hidrológico en estudio. Además, es fundamental contar con datos de calidad, tanto para el entrenamiento como para la validación de los modelos de RNA.

En esta tesis se implementa varios modelos estocásticos y de redes neuronales artificiales para la predicción de escurrimiento medio diario de entrada en la presa La Angostura, Chiapas. Primero, mediante una comparación de modelos de pronóstico Autorregresivo $AR(p)$, Autorregresivo de Promedios Móviles $ARMA(p, q)$, Autorregresivo Integrado de Promedios Móviles (p, d, q) mientras se aumenta el número de parámetros para obtener una mayor precisión del pronóstico y comparar el mejor modelo estocástico con un modelo de redes neuronales artificiales de tipo recurrente. Se usaron datos de caudal medio diario de tres estaciones hidrométricas con periodo común de 1977 a 2007, los primeros 29 años se utilizaron para el entrenamiento y los últimos dos años fueron para pronosticar.

Se evalúa la capacidad de desempeño de estos modelos como herramientas predictivas y tecnológicas para subcuencas que cuentan con estaciones hidrométricas disponibles. Dada la importancia del agua en las actividades cotidianas, es necesario conocer la disponibilidad presente y futura para una adecuada planeación y gestión de

los recursos hídricos. Finalmente, se eligió el modelo de red neuronal recurrente LSTM (Long Short Term Memory) como el mejor modelo para pronosticar escurrimiento medio diario de entrada al embalse, es capaz de pronosticar los caudales medio diario de entrada en la presa del pasado 730 días.

Capítulo 1. GENERALIDADES DE REDES NEURONALES ARTIFICIALES. Se expone un reseño histórico de la red neuronal, se presenta de manera general las redes neuronales: origen y características de las RNA, su estructura básica, varias definiciones según diferentes autores, ventajas y aplicaciones de las redes neuronales y sus usos en el campo de Hidroinformática (medio ambiente).

Capítulo 2. MARCO TEÓRICO. Se presenta el desarrollo teórico de las redes neuronales artificiales tales como: arquitectura de las RNA, sus tipologías, diferentes tipos de funciones de activación, modelos de aprendizaje y algoritmos. Y también definiciones de modelos estocásticos tales como AR, ARMA y ARIMA, proceso estocástico, detección de ruido blanco, prueba de estacionariedad y predicción.

Capítulo 3. METODOLOGÍA. Este capítulo está orientado a la implementación de varios modelos de RNA utilizando datos hidrométricos de la zona de estudio para predicción de escurrimiento medio diario. También Aplicación y comparación de modelos estocásticos AR, ARMA y ARIMA para pronóstico de caudal de entrada en el embalse. Evaluación de la eficiencia de los resultados para encontrar los modelos óptimos estocástico y de red neuronal.

Capítulo 4. DISCUSIONES DE RESULTADOS. En el capítulo 4, se recopilan los resultados obtenidos del capítulo anterior y se hace un análisis, comparación y discusión de los resultados de los modelos de predicción.

CONCLUSIONES Y RECOMENDACIONES. Finalmente, se redacta las conclusiones del trabajo de tesis y se concluye diciendo cuál es el mejor modelo para pronosticar. También se sugieren o recomiendan algunos temas que pudieran necesitar mayor estudio y se los presentan como trabajos a futuro.

CAPÍTULO 1:

1. GENERALIDADES DE REDES NEURONALES ARTIFICIALES

1.1. ANTECEDENTES

Al principio de la década de los cuarenta del siglo XX, la comunidad científica, empezó a estudiar el funcionamiento del cerebro humano, por lo cual surgió la Inteligencia Artificial (IA) se refiere a la capacidad de las máquinas y los sistemas informáticos para imitar o simular ciertas características de la inteligencia humana, y con ello dio origen a las Redes Neuronales Artificiales (RNA) que permiten simular matemáticamente los sistemas naturales utilizando estructuras análogas a las Redes Neuronales Biológicas (RNB).

Las RNA son modelos inspirados en el funcionamiento del cerebro humano. Estas redes están compuestas por nodos interconectados llamados "neuronas artificiales" que procesan y transmiten información. Las redes neuronales se utilizan para reconocimiento de imágenes, procesamiento del lenguaje natural, pronóstico de series temporales y otras aplicaciones de IA. Su primer antecedente data de mediados del siglo XX y las primeras que se conocieron fueron las redes Perceptrón y Adaline (Freeman et al., 1991).

En 1943, el neurólogo **Warren S. McCulloch** y el matemático **Walter Pitts**, los dos pioneros de neurociencia computacional de Estados Unidos, publicaron el artículo "Un cálculo lógico de las ideas inmanentes en la actividad nerviosa" (*A logical calculus of the ideas immanent in nervous activity en inglés*) en el cual modelaron una red neuronal simple mediante circuitos eléctricos. Este artículo fue la base y el inicio para desarrollar las computadoras digitales y (IA) (Frank Rosenblatt inventó la red llamada Perceptrón).

En 1949 el Psicólogo canadiense **Donald Olding Hebb** inventó una hipótesis de aprendizaje que basa en el mecanismo de plasticidad neuronal hoy en día se conoce como aprendizaje de Hebb. Fue el primero en explicar los procesos del aprendizaje desde una perspectiva psicológica, se desarrolló una regla explicando como se ocurría

el aprendizaje. Todavía hoy en día, se encuentra esta regla en un gran número de funciones de aprendizaje en diferentes redes neuronales. Esta teoría proporcionó una explicación temprana de cómo las conexiones neuronales se fortalecen o debilitan en función de la actividad simultánea de las células nerviosas. En otras palabras, la regla de Hebb establece que cuando dos neuronas se activan simultáneamente, la conexión sináptica entre ellas se fortalece.

En 1950, el psicólogo estadounidense **Karl Lashley**, en sus ensayos clínicos, llegó a la conclusión de que la memoria no está localizada en una sola región del cerebro, sino que es distribuida y ampliamente dispersa en múltiples áreas. Esta conclusión fue conocida como la "ley de la equitabilidad de la difusión" o la "hipótesis del engrama masivo". Según Lashley, la pérdida de una parte específica del cerebro no resultaba en la pérdida completa de una memoria o habilidad, sino que más bien la habilidad se deterioraba gradualmente en proporción a la cantidad de tejido cerebral eliminado.

En 1956, se realizó la **primera conferencia de la Inteligencia artificial** en la ciudad de Dartmouth, College en Hanover, New Hampshire, Estados Unidos. Frecuentemente se menciona este Congreso como el nacimiento de la inteligencia artificial y el primer contacto serio con las redes neuronales artificiales. La conferencia fue organizada por John McCarthy, Marvin Minsky, Nathaniel Rochester y Claude Shannon, quienes eran destacados científicos y matemáticos en sus respectivas áreas. El objetivo de la conferencia era reunir a investigadores interesados en el estudio de la inteligencia y explorar la posibilidad de crear máquinas que pudieran exhibir comportamientos inteligentes.

En 1957 **Frank Rosenblatt**, psicólogo estadounidense notable en el campo de la inteligencia artificial, es conocido por su trabajo pionero en el campo de las redes neuronales y, en particular, por desarrollar el "**Perceptrón**", uno de los primeros modelos de red neuronal artificial. Fue un sistema de clasificador e identificador de patrones geométricos y abstractos.

El perceptrón consistía en unidades de procesamiento interconectadas (neuronas) que podían aprender y tomar decisiones basadas en entradas. Sin embargo, sus afirmaciones sobre las capacidades del perceptrón fueron posteriormente criticadas por Marvin Minsky y Seymour Papert, quienes demostraron las limitaciones del perceptrón para resolver problemas más complejos. Este revés, conocido como la "controversia del perceptrón", disminuyó temporalmente el entusiasmo por las redes neuronales en la comunidad de IA.

En 1959, **Bernard Widrow**, profesor estadounidense de ingeniería eléctrica en la Universidad de Stanford de Estados Unidos y coinventor junto con Marcial Hoof, su estudiante de doctorado, desarrollaron un algoritmo adaptativo lineal llamado "**ADALINE**" (ADaptive LINear Elements). Este algoritmo es ampliamente utilizado en aplicaciones de procesamiento de señales, como la cancelación de eco y el equalizador adaptativo. A lo largo de su carrera, Widrow ha realizado importantes investigaciones en el procesamiento adaptativo de señales y las redes neuronales. Sus contribuciones continúan siendo relevantes en la actualidad y han influido en el desarrollo de la inteligencia artificial y el aprendizaje automático.

En 1969, **Marvin Lee Minsky** fue un destacado científico de la computación y pionero en el campo de la inteligencia artificial. Durante su carrera, fue profesor en el MIT (Instituto de Tecnología de Massachusetts) y cofundador del Laboratorio de Inteligencia Artificial del MIT. Junto con Seymour Papert inventaron el lenguaje de programación Logo en 1968. En el año 1969 casi se llegó a producir la desaparición abrupta de las RN porque **Minsky y Papert** matemáticamente demostraron que los Perceptrones eran incapaces de solucionar problemas relativamente fáciles, tales como el aprendizaje de una función no lineal. Entonces llegaron a la conclusión que el Perceptrón y el tema neuro computacional no eran interesantes para estudiar y desarrollar. A partir de este momento se ha reducido de manera considerable las inversiones en el campo de la investigación neuro computacional, entonces muy pocos investigadores ha continuado con sus trabajos de investigación sobre las RNA.

Seria hasta en 1982 **John Joseph Hopfield**, físico y neurocientífico estadounidense más conocido por su invención de una red neuronal asociativa conocida como la red de Hopfield, que son modelos matemáticos inspirados en el funcionamiento del cerebro humano, con la publicación del artículo "*Redes neuronales y sistemas físicos con habilidades computacionales colectivas emergentes*", Hopfield inventó el algoritmo Backpropagation entonces devolvió el interés a los investigadores sobre los trabajos de investigación en el campo de la computación neuronal después de dos décadas de inactividad. También en 1985 publicó su libro: "*Computación neuronal de decisiones en problemas de optimización*" que ha renacido las redes neuronales.

A partir de 1986 **hasta la fecha**, las redes neuronales han experimentado un rápido desarrollo y han sido objeto de intensa investigación y aplicación en una amplia gama de campos tales como aprendizaje profundo (Deep Learning), redes neuronales convolucionales (CNN), redes neuronales recurrentes (RNN) y redes generativas adversariales (GAN). Las redes neuronales se han aplicado con éxito en campos como la medicina, finanzas, hidrología, robótica, sistemas de recomendación, procesamiento de voz y muchos otros. Su capacidad para aprender patrones complejos y extraer información relevante de grandes conjuntos de datos ha impulsado avances en estas áreas.

Hoy en día, las redes neuronales y el aprendizaje profundo son áreas activas de investigación y aplicación en diversos campos, como la visión por computadora, la robótica, la medicina, las finanzas y muchos otros, impulsando avances significativos en la inteligencia artificial.

1.2. PLANTEAMIENTO DEL PROBLEMA

El pronóstico del tiempo se refiere a una predicción o estimación de las condiciones climáticas esperadas en un área geográfica específica durante un período de tiempo determinado. Esto incluye información sobre variables climáticas como temperatura, humedad, viento, precipitación, nubosidad y otras condiciones atmosféricas relevantes. Durante siglos el mundo ha tenido predicciones relacionadas de cómo sería el clima por

un día, una semana y hasta por una estación completa. Pero la mayoría de los métodos predictivos solamente tomaban en cuenta los patrones de señalamiento de vientos, pero dichas predicciones no se llegaron a cumplir y tampoco tenían la posibilidad de realizar pruebas estadísticas para comprobar la certeza de estos eventos.

Uno de los problemas fundamentales en la Hidrología es la determinación de modelos que permitan pronosticar con oportunidad el volumen de escurrimiento proveniente de una avenida que podría presentarse ante una tormenta ocurrida a partir de los valores de precipitación en instantes previos al de interés y otro problema es la transformación del proceso lluvia-escurrimiento en una cuenca, ya que involucra varios parámetros espaciotemporales (Domínguez et al., 2004).

En México, desde hace algunos años la Comisión Nacional del Agua tiene el interés en la modernización de las estaciones hidroclimatológicas para la estimación de los grandes volúmenes de agua que llegan hacia las presas. Sin embargo, el problema de administración del recurso hídrico, determinación del nivel de aguas máximas sobre todo en temporada de lluvia y pronósticos de volumen de escurrimientos máximos diarios en presas hidroeléctricas en el país no son muy frecuente hoy en día.

1.3. JUSTIFICACIÓN

Según el índice de riesgo climático global (IRC) entre 2000 a 2019 murieron 475 mil personas en total como consecuencia directa de más de 11 mil fenómenos meteorológicos extremos y las pérdidas económicas ascendieron a unos 2.56 billones de dólares estadounidenses a nivel mundial (IRC, 2021).

En México según los datos de CONAGUA (La Comisión Nacional del Agua) en el año de 2017 solamente contaba de 3 079 estaciones climatológicas que estaban en operación para la medición de variables hidroclimatológicas, de estas estaciones solamente 81 eran capaces de transmitir la información meteorológica en tiempo real. Estas estaciones median variables hidroclimatológicas tales como caudales, lluvias, velocidad y dirección del viento, precipitación, evaporación y temperatura. En México

se dispone de 855 estaciones hidrométricas, entre ellas algunas automáticas (CONAGUA, 2018).

Con la necesidad de tener herramientas tecnológicas para realizar predicciones espaciotemporales y debido al gran uso de las redes neuronales artificiales en problemas de predicción de variables hidroclimatológicas como precipitación, caudales y el proceso transformación lluvia - escurrimiento.

Entre las principales características de las RNA que es un modelo con múltiples parámetros, el cual es capaz de reproducir complejas relaciones no lineales, cuyo proceso de calibración (entrenamiento) requiere de gran cantidad de información, siendo el modelo resultante veloz y que puede ser utilizado donde los modelos físicos resultan inadecuados o donde pueda complementarlos (Solomatine, 2002).

Tomando en cuenta la información anterior, en este estudio se decide utilizar varios modelos para predicción de volumen de escurrimiento medio diario a partir de datos de caudales medios diarios de estaciones hidrométricas mediante modelos estocásticos y RNA aplicada a la presa La Angostura, Chiapas, México. A fin de tener una herramienta de pronóstico para determinar el nivel de aguas máximas según los volúmenes de escurrimientos medios diarios pronosticados en la presa.

1.4. OBJETIVO GENERAL

Predecir volumen de escurrimiento medio diario en la presa La Angostura, Chiapas utilizando redes neuronales artificiales y modelos autorregresivos AR, ARMA y ARIMA.

1.5. OBJETIVOS ESPECÍFICOS

Desarrollar modelos predictivos para determinar el volumen de escurrimientos medio diario mediante modelos autorregresivos AR, ARMA, ARIMA y de redes neuronales artificiales a partir de datos de caudales de las estaciones hidrométricas de la zona de estudio.

Entrenar la red neuronal mediante modelos supervisados con registros de datos hidrométricos y comparar los resultados obtenidos en las neuronas de salida con la salida que se desea obtener y con los modelos autorregresivos.

Calcular el desempeño de los modelos autorregresivos y los de las redes neuronales como instrumento para modelizar y predecir el volumen de escurrimientos medio diario en la presa La Angostura.

1.6. HIPÓTESIS

Los modelos de pronóstico de volumen de escurrimientos medio diario mediante Redes Neuronales Artificiales tienen la capacidad para identificar el patrón del comportamiento de los caudales de las estaciones hidrométricas con relaciones no lineales de las variables; son una herramienta predictiva eficiente y tecnológica de bajo costo para la estimación de escurrimientos superficiales y nivel máximo de aguas para aprovechamientos hidroeléctricos.

1.7. ORIGEN DE LAS REDES NEURONALES

Las redes neuronales tienen su origen en la ciencia y la ingeniería, inspiradas en el funcionamiento del cerebro humano y en la forma en que las neuronas interactúan entre sí para procesar información. Buscando la capacidad para el diseño y construcción de máquinas inteligentes y capaces de realizar procesos complejos fueron los objetivos principales de los científicos a lo largo de la historia de las redes neuronales.

Las redes neuronales que son parte de la inteligencia artificial se basan en el desarrollo de algoritmos y modelos que permiten a las máquinas aprender, razonar, tomar decisiones y resolver problemas de manera similar a como lo haría un ser humano. El campo de la inteligencia artificial va más allá, no sólo intenta comprender, sino que también se esfuerza en construir entidades inteligentes. (Russell et al., 2004).

Las redes neuronales artificiales son modelos computacionales inspirados en el funcionamiento del sistema nervioso humano. Estas redes están compuestas por

unidades de procesamiento interconectadas, conocidas como neuronas artificiales o nodos, que trabajan juntas para procesar información y realizar tareas específicas. Las redes neuronales artificiales se han utilizado con éxito en una amplia gama de aplicaciones, incluyendo reconocimiento de patrones, clasificación de imágenes, procesamiento del lenguaje natural, traducción automática, análisis de datos, robótica y muchos otros campos donde el aprendizaje y la adaptabilidad son necesarios.

Alan Turing, fue el primero en estudiar el cerebro humano como una forma de ver el mundo de la computación mediante lo cual ha propuesto la prueba de Turing en 1950, consiste en que un computador supera la prueba si un evaluador humano no es capaz de distinguir si las respuestas, a una serie de preguntas planteadas, son de una persona o no (Maren et al., 1990). Las redes neuronales artificiales pueden tener diferentes arquitecturas y topologías, dependiendo de la tarea que se esté abordando.

CAPÍTULO 2:

2. MARCO TEÓRICO

2.1. CARACTERÍSTICAS DE LAS REDES NEURONALES ARTIFICIALES

Las Redes Neuronales Artificiales, (en inglés Artificial Neural Networks ANN) tienen varias características distintivas que las hacen poderosas y versátiles para abordar problemas complejos. Algunas de las características clave de las redes neuronales artificiales tales como aprendizaje adaptativo, capacidad de generalización, tolerancia al ruido, paralelismo y procesamiento distribuido, adaptabilidad y plasticidad, extracción automática de características. Estas características hacen que las redes neuronales artificiales sean herramientas poderosas para abordar problemas complejos de aprendizaje automático, reconocimiento de patrones y procesamiento de datos. Su capacidad para aprender, adaptarse y generalizar a partir de los datos las ha convertido en una tecnología clave en el campo de la inteligencia artificial (Hilera, 1995).

Las redes neuronales pueden ajustar los pesos y las conexiones entre las neuronas en función de los ejemplos de entrenamiento, lo que les permite mejorar su rendimiento a medida que se les presenta más información. Dado que las neuronas en una red trabajan en paralelo y están conectadas entre sí, pueden compensar errores o anomalías en los datos de entrada, lo que las hace adecuadas para aplicaciones en entornos ruidosos o con datos incompletos. Algunas RNA son capaces de abstraer la esencia de un conjunto de entradas que aparentemente no presentan aspectos comunes o relativos (Viñuela, 2004).

2.2. ESTRUCTURA BÁSICA DE UNA RED NEURONAL

2.2.1. Analogía con el Cerebro

La estructura básica de una red neuronal consta de tres componentes principales: capas, neuronas y conexiones. La estructura básica de una red neuronal incluye capas

de neuronas interconectadas mediante conexiones ponderadas. Esta estructura permite que la red procese información, aprenda a partir de los datos y realice tareas específicas, como clasificación, reconocimiento de patrones o predicciones.

Una forma común de analizar las redes neuronales es mediante una analogía con el cerebro humano. Aunque las redes neuronales artificiales no replican exactamente la complejidad y funcionalidad del cerebro humano, la analogía puede ayudar a comprender algunos aspectos básicos.

En una red neuronal artificial, las neuronas artificiales se inspiran en las neuronas biológicas del cerebro. Al igual que las neuronas biológicas, las neuronas artificiales reciben señales de entrada, las procesan y generan una señal de salida. Sin embargo, las neuronas artificiales son simplificaciones matemáticas de las neuronas biológicas y no tienen la misma complejidad y funcionalidad. La **Figura 2.1** muestra las partes que constituyen una neurona.

Es importante destacar que las redes neuronales artificiales son simplificaciones matemáticas y computacionales de los procesos biológicos del cerebro. Aunque la analogía puede ser útil para entender algunos aspectos básicos, las redes neuronales artificiales están diseñadas con un enfoque específico en la solución de problemas computacionales y no reflejan completamente la complejidad y la funcionalidad del cerebro humano.

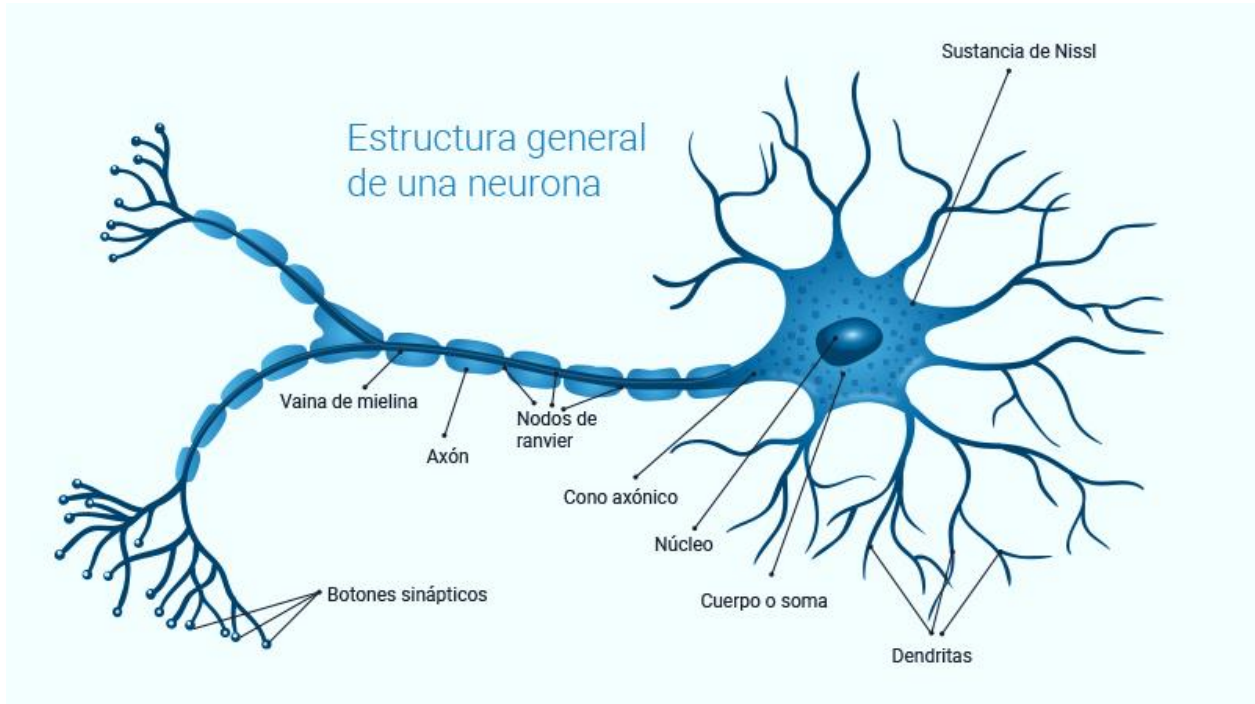


Figura 2.1. Estructura general de una neurona biológica¹.

El número exacto de neuronas en el cerebro humano no se conoce con precisión, ya que puede variar de una persona a otra. Sin embargo, se estima que el cerebro humano adulto contiene alrededor de 86 mil millones (86 000 000 000) de neuronas. Estas neuronas están distribuidas en diferentes regiones del cerebro y se interconectan a través de miles de millones de sinapsis, formando una red extremadamente compleja.

Es importante destacar que las redes neuronales artificiales utilizadas en la inteligencia artificial también pueden variar en el número de neuronas, dependiendo de la tarea y la complejidad del problema que se esté abordando. En algunos casos, las redes pueden tener solo unas pocas docenas de neuronas, mientras que en otros casos pueden tener millones o incluso miles de millones de neuronas artificiales.

¹ Figura toma de: Inspiración biológica de las redes neuronales artificiales.
<https://medium.com/soldai/inspiraci%C3%B3n-biol%C3%B3gica-de-las-redes-neuronales-artificiales-9af7d7b906a> [Consultado: 18/09/2022].

2.2.2. Redes Neuronales Artificiales

Las redes neuronales artificiales son modelos matemáticos y computacionales inspirados en el funcionamiento del sistema nervioso biológico. Estas redes están compuestas por unidades de procesamiento llamadas neuronas artificiales o nodos, que se organizan en capas y se interconectan entre sí mediante conexiones ponderadas. Las redes neuronales artificiales son flexibles y pueden adaptarse a diferentes tipos de datos y problemas. Su capacidad para aprender a partir de ejemplos y generalizar a partir de los datos las ha convertido en una herramienta poderosa en el campo de la inteligencia artificial y han demostrado un gran éxito en una amplia variedad de aplicaciones.

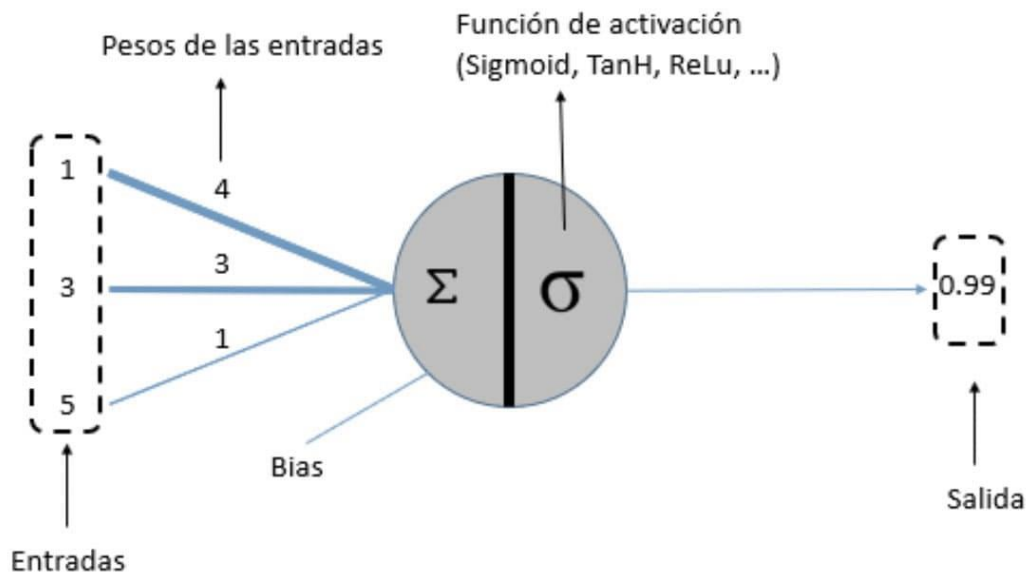


Figura 2.2. Diagrama de una neurona artificial².

² Figura tomada de: Redes Neuronales artificiales: Qué son y cómo se entrenan. En la página de <https://www.xeridia.com/blog/redes-neuronales-artificiales-que-son-y-como-se-entrenan-parte-i> [Consultado: 18/09/2022]

La **Figura 2.1** muestra las partes de una neurona artificial que son los datos de entrada, la ponderación asignada a las entradas, función de activación, el residual y el valor a la salida.

2.3. DEFINICIONES DE UNA RED NEURONAL

Hay diversas definiciones de las redes neuronales artificiales partiendo con definiciones cortas a generales para intentar explicar el funcionamiento de las redes neuronales artificiales con más detalles.

Por ejemplo:

- *Una nueva forma computacional, que se inspira en modelos biológicos.*
- *Un modelo matemático que este compuesto por diferentes elementos procesales que están organizados a diferentes niveles.*
- *"... un sistema de computación compuesto por un gran número de elementos simples, elementos de procesos muy interconectados, los cuales procesan información por medio de su estado dinámico como respuesta a entradas externas" (Hecht-Niesen, 1987).*
- *"Redes neuronales artificiales son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico" (Kohonen, 1988).*
- *Redes neuronales artificiales son una representación técnica de modelación matemática, que es un intento de la imitación del proceso de aprendizaje de lo que ocurre en el sistema nervioso.*

2.4. VENTAJAS QUE OFRECEN LAS REDES NEURONALES

Las redes neuronales ofrecen varias ventajas y beneficios que las hacen atractivas para su uso en diversas aplicaciones. Si bien las redes neuronales ofrecen muchas ventajas, también es importante tener en cuenta que su desempeño puede depender en gran medida de la cantidad y calidad de los datos de entrenamiento, la elección adecuada de

la arquitectura de la red y los parámetros de entrenamiento. Además, las redes neuronales pueden ser computacionalmente intensivas y requerir recursos significativos para su entrenamiento y ejecución. Esto hace que ofrezcan numerosas ventajas y que este tipo de tecnología se esté aplicando en múltiples áreas (Maren et al.,1990).

Entre las ventajas se incluyen:

➤ **Aprendizaje Adaptativo**

El cerebro humano tiene la capacidad de aprender y adaptarse a partir de la experiencia, y se cree que esto ocurre a través de cambios en las conexiones sinápticas. Las redes neuronales artificiales también tienen la capacidad de aprender y adaptarse mediante el ajuste de los pesos de las conexiones entre las neuronas. Esto se asemeja a la plasticidad sináptica en el cerebro, donde las conexiones se fortalecen o debilitan en función de la experiencia y el aprendizaje. Pueden descubrir patrones complejos, extraer características relevantes y generalizar a partir de ejemplos de entrenamiento. Esto les permite abordar problemas difíciles en los que las soluciones tradicionales pueden ser difíciles de diseñar o implementar.

➤ **Autoorganización**

Se refiere a la capacidad de la red para organizar y estructurar sus conexiones y neuronas de manera automática, sin la necesidad de una supervisión externa o un diseño manual detallado. En otras palabras, las redes neuronales tienen la capacidad de aprender y descubrir estructuras y patrones en los datos de forma autónoma. La autoorganización en las redes neuronales puede ocurrir a través de diferentes mecanismos y modelos, y es especialmente relevante en el campo de las redes neuronales no supervisadas.

➤ **Tolerancia a fallos**

Las redes neuronales pueden exhibir cierto grado de tolerancia a fallos, lo que significa que pueden continuar funcionando y produciendo resultados aceptables incluso cuando algunas de sus neuronas o conexiones presentan fallas o están dañadas. Sin embargo,

la tolerancia a fallos en las redes neuronales puede variar dependiendo de la arquitectura y el tipo de red utilizada.

Aunque las redes neuronales pueden tener cierta tolerancia a fallos, es importante tener en cuenta que no son inmunes a fallas graves o masivas. Las fallas críticas en una red neuronal, como la corrupción generalizada de datos o la pérdida de un gran número de neuronas, pueden afectar gravemente su capacidad para funcionar correctamente.

En general, la tolerancia a fallos en las redes neuronales puede depender de varios factores, como la arquitectura de la red, la redundancia incorporada, el tipo de falla y el grado de daño. La investigación en este campo continúa avanzando para mejorar la robustez y la tolerancia a fallos de las redes neuronales en diferentes aplicaciones y escenarios.

La razón por la que las redes neuronales son tolerantes a los fallos es que tienen su información distribuida en las conexiones entre neuronas, existiendo cierto grado de redundancia en este tipo de almacenamiento (Hilera, 1995).

➤ **Operación en tiempo real**

Las redes neuronales pueden operar en tiempo real, lo que significa que pueden procesar y generar resultados en tiempo casi instantáneo mientras se reciben datos de entrada continuos. La capacidad de operar en tiempo real depende de varios factores, como la arquitectura de la red, el tamaño del modelo, la complejidad de la tarea y la capacidad de procesamiento del sistema en el que se implementa la red neuronal. Algunos aspectos clave relacionados con la operación en tiempo real de las redes neuronales son eficiencia computacional, arquitecturas ligeras, hardware especializado; paralelización y distribución.

Es importante tener en cuenta que la operación en tiempo real de una red neuronal puede estar sujeta a restricciones y limitaciones dependiendo de la tarea y los recursos disponibles. Algunas aplicaciones pueden requerir un procesamiento en tiempo real más estricto, mientras que otras pueden tolerar cierto retraso o latencia. La elección de la arquitectura de la red, el hardware utilizado y las técnicas de implementación

dependerán de los requisitos específicos de cada aplicación y del equilibrio entre precisión y velocidad requerido.

➤ **Fácil inserción dentro de la tecnología existente**

Las redes neuronales artificiales pueden ser fácilmente integradas en la tecnología existente debido a su flexibilidad en la entrada y salida, interfaces de programación amigables, bibliotecas y herramientas existentes, escalabilidad y servicios en la nube disponibles. Esto hace que sea más accesible para los desarrolladores y empresas aprovechar los beneficios del aprendizaje automático y las redes neuronales en sus sistemas existentes.

2.5. APLICACIONES DE LAS REDES NEURONALES

Estas son solo algunas de las muchas aplicaciones de las redes neuronales. Con el avance continuo de la tecnología y la investigación, se están explorando nuevas áreas de aplicación para aprovechar las capacidades de las redes neuronales en diversas industrias y campos. Las redes neuronales tienen una amplia gama de aplicaciones en diversos campos.

Algunas aplicaciones comerciales son (Hilera, 1995):

A. Biología:

- Aprendizaje cercano del funcionamiento cerebral y demás sistemas.
- Obtención de modelización de la retina.

B. Empresa:

- Evaluación de posibles formaciones geológicas y petrolíferas.
- Identificación de candidatos para ocupación de plazas específicas.
- Explotación de bases de datos.
- Optimización de plazas y horarios en líneas de vuelo.
- Optimización del flujo del tránsito en las carreteras mediante el control de los semáforos.
- Reconocimiento de caracteres escritos.

- Modelación de sistemas para automatizar y controlar sistemas.

C. Medio ambiente:

- Análisis de tendencias y patrones climatológicos.
- Previsión y predicción del tiempo.

D. Finanzas:

- Previsión de la evolución de los precios en los mercados.
- Valoración del riesgo de los créditos.
- Identificación de falsificaciones.
- Interpretación de firmas.

E. Manufacturación:

- Robots automatizados y sistemas de control.
- Control de producción en líneas de procesos.
- Inspección de la calidad.

F. Medicina:

- Analizadores del habla para ayudar en la audición de sordos profundos.
- Diagnóstico y tratamiento a partir de síntomas y/o de datos analíticos.
- Monitorización en cirugías.
- Predicción de reacciones adversas en los medicamentos.
- Entendimiento de la causa de los ataques cardíacos.
- Lectores de rayos X.

G. Militares:

- Clasificación de las señales de radar.
- Creación de armas inteligentes.
- Optimización del uso de recursos escasos.
- Reconocimiento y seguimiento en el tiro al blanco.

Las redes neuronales se utilizan en sistemas de control y automatización para modelar y controlar procesos complejos, como sistemas de control de tráfico, sistemas de control de producción, sistemas de control de energía, entre otros. Se aplican en análisis financiero, pronóstico de precios de acciones, predicción de tendencias del mercado y detección de fraudes financieros.

Estas son solo algunas de las muchas aplicaciones de las redes neuronales. La versatilidad y el poder de las redes neuronales las hacen útiles en una amplia variedad de campos y sectores, impulsando avances significativos en tecnología, medicina, industria y más.

2.6. ARQUITECTURA DE LAS REDES NEURONALES

Las redes neuronales artificiales están compuestas por una serie de capas interconectadas de neuronas artificiales, y la arquitectura de una red neuronal se refiere a la estructura y organización de estas capas y conexiones. Hay diferentes tipos de arquitecturas de redes neuronales que se utilizan para diferentes tipos de tareas y problemas. Hay muchas otras arquitecturas y variantes que se han desarrollado para abordar problemas específicos o mejorar el rendimiento en determinadas tareas. La elección de la arquitectura adecuada depende del problema que se esté abordando y de los datos disponibles (Russell et al., 2004).

2.6.1. Clasificación por tipo de capas

Se distinguen tres tipos de capas (**Figura 2.3**):

➤ Capa de entrada

La capa de entrada es la primera capa de una red neuronal y su función principal es recibir los datos de entrada y transmitirlos a las capas ocultas de la red. Cada nodo en la capa de entrada representa una característica o atributo del dato de entrada.

La capa de entrada de una red neuronal es la primera capa que recibe los datos de entrada y consta de nodos que representan las características o atributos de los datos. Su función es pasar los valores de entrada a las capas ocultas de la red para su procesamiento posterior.

➤ Capa de salida

La capa de salida es la última capa de una red neuronal y su función principal es generar la salida final de la red en respuesta a los datos de entrada. La cantidad de nodos en la capa de salida depende del tipo de problema que se esté abordando y del tipo de salida deseada. La capa de salida es donde se obtiene la respuesta final de la red neuronal en función de los datos de entrada y los pesos y bias aprendidos durante el entrenamiento.

➤ Capas ocultas

La capa oculta en una red neuronal se encuentra entre la capa de entrada y la capa de salida. Su nombre proviene del hecho de que las neuronas en esta capa no están directamente conectadas a las entradas ni a las salidas externas de la red, por lo tanto, su funcionamiento y procesamiento no son visibles externamente. La cantidad de capas ocultas y el número de neuronas en cada capa oculta pueden variar según la arquitectura y complejidad de la red neuronal.

Este tipo de capa oculta proporciona grados de libertad a la red neuronal gracias a los cuales es capaz de representar más fehacientemente determinadas características del entorno que trata de modelar (Torres, 2020).

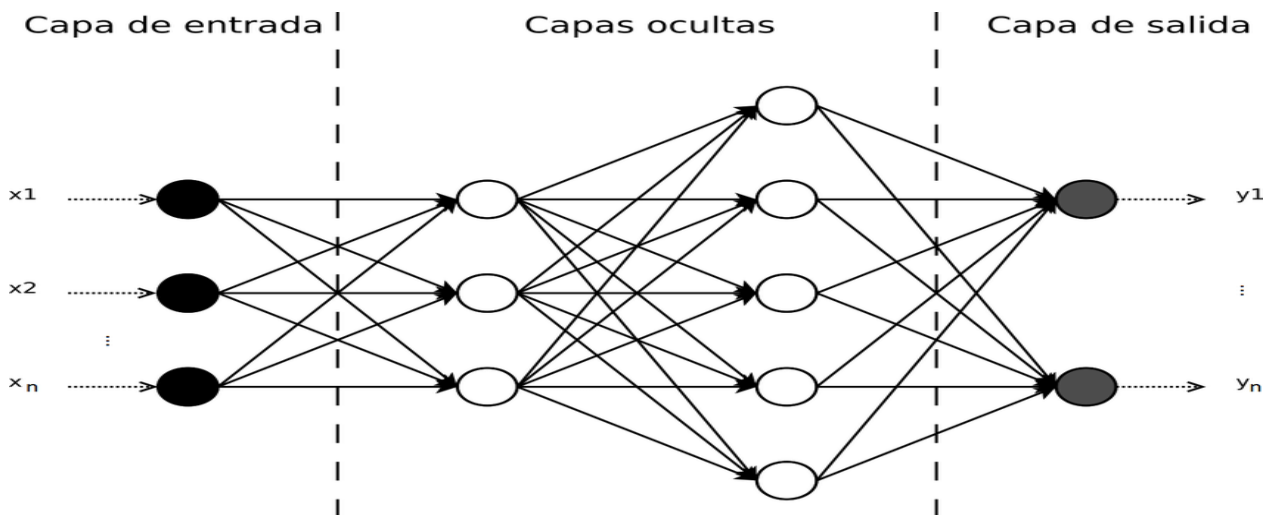


Figura 2.3. Arquitectura unidireccional de una red neuronal (Meneses, 2017).

2.6.2. Clasificación por número de capas

Dependiendo de la estructura de las capas de la red neuronal

Figura 2.4. se puede hablar de:

➤ **Redes monocapa**

Como su nombre lo indica contiene solamente una capa de neuronas. En las redes neuronales monocapa, las conexiones entre los nodos de la capa de entrada y la capa de salida están ponderadas por pesos, que determinan la influencia de cada nodo de entrada en los nodos de salida. Además, cada nodo de salida puede tener una función de activación asociada para generar la salida final de la red.

➤ **Redes multicapa**

Las redes neuronales multicapa, también conocidas como redes neuronales de múltiples capas, son un tipo de arquitectura de redes neuronales que consta de una o más capas ocultas entre la capa de entrada y la capa de salida. A diferencia de las redes neuronales monocapa, las redes multicapa tienen una mayor capacidad para aprender relaciones y patrones complejos en los datos.

Las redes neuronales multicapa son ampliamente utilizadas en una variedad de aplicaciones de aprendizaje automático y procesamiento de datos, incluyendo clasificación de imágenes, reconocimiento de voz, procesamiento del lenguaje natural, predicción y muchas otras tareas donde se requiere un modelado más complejo y una capacidad de generalización mejorada.

Teniendo en cuenta el flujo de datos, se puede distinguir:

➤ **Redes unidireccionales (*feedforward*)**

Las redes neuronales unidireccionales, también conocidas como redes feedforward, son un tipo de arquitectura de redes neuronales donde la información fluye en una

dirección, desde la capa de entrada hacia la capa de salida, sin bucles o retroalimentación de información.

➤ **Redes recurrentes o realimentadas (*feedback*)**

Las redes neuronales recurrentes (RNN, por sus siglas en inglés) o redes realimentadas son un tipo de arquitectura de redes neuronales en las que las conexiones entre las neuronas forman bucles, lo que permite que la información fluya en retroalimentación o en ciclos a través de la red. Esta característica las distingue de las redes neuronales unidireccionales (feedforward), en las que la información fluye solo en una dirección, de la entrada hacia la salida.

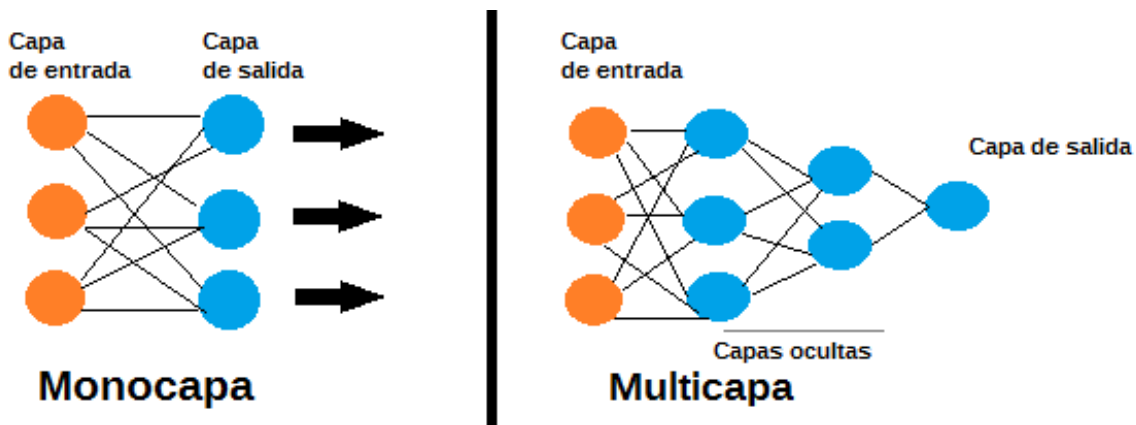


Figura 2.4. Arquitectura de redes monocapa y multicapa³.

2.7. MODELO DE UNA RED NEURONAL

El modelo de una red neuronal se refiere a la representación y estructura de una red neuronal específica. En todo modelo artificial de neurona se tienen cuatro elementos básicos, la **Figura 2.5** muestra las diferentes partes de una red neuronal (Solomatine, 2002):

- I. Un conjunto de conexiones ponderadas, cada conexión entre los nodos de las capas está asociada a un peso. Estos pesos determinan la influencia de cada nodo en la capa anterior en los nodos de la capa siguiente. Durante el

³ Figura tomada de: Tipos de redes neuronales (Clasificación). <https://inteligencia-artificial.dev/tipos-redes-neuronales/> [Consultado: 20/09/2022]

- entrenamiento de la red neuronal, los pesos se ajustan para mejorar el rendimiento de la red.
- II. Algoritmo de aprendizaje para entrenar la red neuronal y ajustar los pesos, se utiliza un algoritmo de aprendizaje.
 - III. Funciones de activación, cada nodo en la red neuronal aplica una función de activación a la suma ponderada de las entradas recibidas. La función de activación introduce la no linealidad en la red y determina la salida o activación del nodo.
 - IV. Función de pérdida es una medida que se utiliza para evaluar el error o la discrepancia entre las salidas predichas por la red neuronal y las salidas esperadas.

Las entradas reciben los datos o parámetros que le permiten decidir a la neurona se estará activa o no, normalmente se presentan como x_1, x_2, \dots, x_n . Las salidas generan una respuesta para cada neurona dependiendo si se activa o no para finalmente representar cada salida, así como y_1, y_2, \dots, y_n .

2.7.1. Entrada y salida

La entrada y/o salida de una red neuronal puede clasificarse en dos grupos, es decir, grupo binario o continuo. Las neuronas binarias, solamente aceptan dos valores posibles (0, 1) o (-1, 1). Por otra parte, las neuronas continuas admiten valores dentro de un rango específico que generalmente varía en el intervalo de (-1, 1). Para seleccionar el modelo de red neuronal eso depende en que campo se va a aplicar el modelo y su arquitectura.

2.7.2. Pesos

En una red neuronal artificial, los pesos son parámetros ajustables que determinan la influencia de cada conexión entre los nodos de la red. Los pesos son asignados a las conexiones entre las neuronas y representan la fuerza o importancia de cada conexión en el procesamiento de la información. Es importante destacar que los pesos son

específicos de cada red neuronal y se determinan durante el entrenamiento de la red. Los pesos aprendidos permiten que la red neuronal capture las relaciones y patrones en los datos, y su ajuste óptimo es crucial para el rendimiento y la precisión de la red en tareas de predicción, clasificación u otras tareas de aprendizaje automático.

2.7.3. Clasificador lineal o regla de propagación

La entrada a un clasificador lineal en dos dimensiones tiene la siguiente forma (ecuación (2.1)):

$$(2.1) \quad f(x_1, x_2) = b + w_1x_1 + w_2x_2$$

Si se lo extiende a cualquier número de dimensiones, se puede expresar como (ecuación (2.2)):

$$(2.2) \quad f(X) = b + \sum_{i=1}^n w_i x_i$$

Para la regresión, $f(X)$ arroja el resultado de la predicción, teniendo como capa de entrada X . Para la clasificación, los valores de predicción se calculan con la ecuación (2.3):

$$(2.3) \quad \text{Clasificación} = \begin{cases} 1 & \text{si } f(X) > 0 \\ 0 & \text{si } f(X) \leq 0 \end{cases}$$

2.7.4. Función de activación

La función de activación o función de transferencia (**Figura 2.5**) se encarga de devolver una salida a partir de un valor de entrada, normalmente el conjunto de valores de salida en un rango determinado como $(0, 1)$ o $(-1, 1)$. La función de activación no lineal ayuda a modelar funciones o curvas no triviales. Se buscan funciones de activación en las que las derivadas sean simples, para minimizar el consumo de los recursos computacionales (Solomatine, 2002).

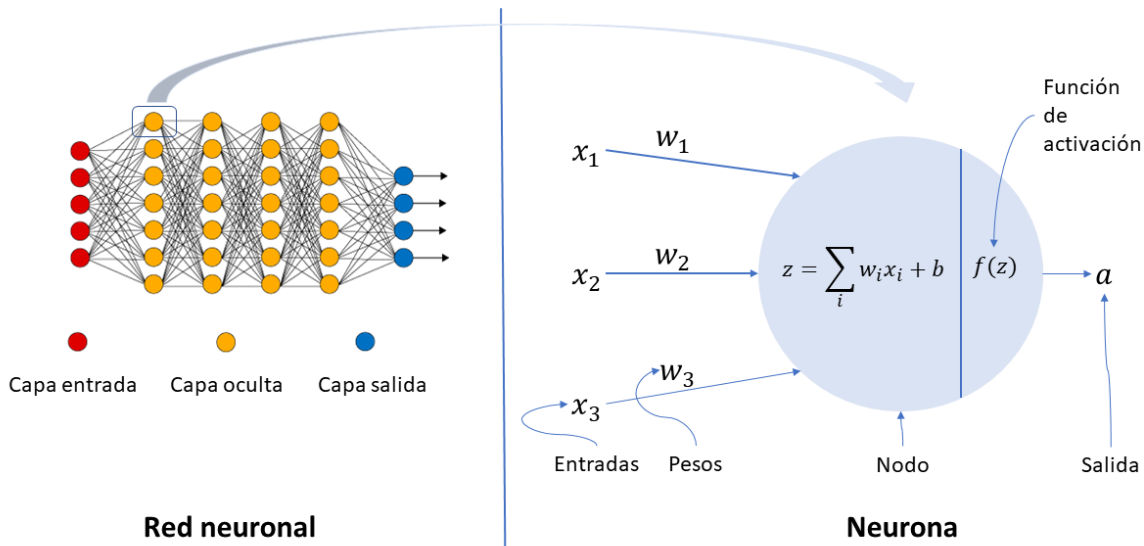


Figura 2.5. Función de activación de una red neuronal⁴.

La función de activación es una función matemática que se aplica a la salida de un nodo o neurona en una red neuronal. Su propósito principal es introducir no linealidad en la red neuronal, permitiendo que la red pueda aprender y representar relaciones y patrones complejos en los datos.

2.7.4.1. Características de función de activación

Las funciones de activación en las redes neuronales tienen varias características que determinan su comportamiento y su idoneidad para diferentes tipos de problemas. Es de suma importancia que toda función de activación cumpla estas características:

➤ No linealidad

Una característica clave de las funciones de activación es su capacidad para introducir no linealidad en la red neuronal.

⁴ Figura toma de: Función de activación. <https://jahazielponce.com/funciones-de-activacion-y-como-puedes-crear-la-tuya-usando-python-r-y-tensorflow/> [Consultado: 22/09/2022]

Esto es esencial para que la red pueda aprender y representar relaciones y patrones complejos en los datos. Las funciones de activación no lineales permiten que la red neuronal modele datos no lineales y capture la no linealidad en los datos de entrada.

➤ **Comportamiento asintótico**

Algunas funciones de activación tienen un comportamiento asintótico a medida que los valores de entrada se vuelven extremadamente grandes o pequeños. Por ejemplo, la función sigmoide se aproxima a 0 o 1 a medida que los valores de entrada tienden hacia menos infinito o más infinito, respectivamente.

➤ **Rango y escala**

Las funciones de activación pueden tener diferentes rangos y escalas. Es importante considerar el rango y la escala de la función de activación en relación con los datos y el problema que se está abordando.

➤ **Derivabilidad**

Es importante en el proceso de entrenamiento de la red neuronal, ya que se utiliza el cálculo del gradiente para ajustar los pesos y minimizar la función de pérdida. Las funciones de activación deben ser derivables en la mayoría de los puntos para permitir el uso de algoritmos de optimización basados en el gradiente. Esto permite la retropropagación del error y el aprendizaje eficiente de la red neuronal.

2.7.4.2. Tipos de función de activación

Hay varias funciones de activación comunes utilizadas en las redes neuronales. Estas son algunas de las más utilizadas:

I. Sigmoidea

La función sigmoide (también conocida como función logística) tiene la forma de una curva en forma de "S". Toma cualquier valor de entrada y lo mapea a un rango (0,1). Históricamente, la función sigmoidea es la función de activación más antigua, se define como la ecuación (2.6:

$$(2.4) \quad f(x) = \frac{1}{1-e^{-x}}$$

Con la ecuación de la suma ponderada de entrada:

$$(2.5) \quad z = b + \sum_{i=1}^n w_i x_i$$

Entonces:

$$(2.6) \quad f(z) = \frac{1}{1-e^{-z}}$$

Características:

Saturación y eliminación del gradiente.

Se converge lentamente.

No tiene asíntota en valor de cero.

Se corta entre los valores de 0 y 1.

Tiene un buen desempeño en la capa final.

II. Tangente hiperbólica

La función tanh es similar a la función sigmoidea, pero tiene un rango entre $(-1, 1)$. La función tanh es útil en redes neuronales donde las salidas deben tener valores negativos y positivos.

$$(2.7) \quad f(z) = \frac{2}{1+e^{-2z}} - 1$$

Características:

Es muy parecida a la función de sigmoidea

Saturación y eliminación del gradiente.

Se converge lentamente.

Se centra en valor de 0.

Se corta entre los valores de -1 y 1 .

Se utiliza mucho en clasificación binaria.

Buenos resultados en modelos recurrentes.

III. ReLU – Rectified Lineal Unit

La función ReLU es una función no lineal que devuelve cero para cualquier valor de entrada negativo y el mismo valor de entrada para cualquier valor positivo. Se puede definir como:

$$(2.8) \quad f(z) = \max(0, z) = \begin{cases} 0 & \text{para } z < 0 \\ z & \text{para } z \geq 0 \end{cases}$$

Sus características:

No se corta en ningún valor.

Se podría desaparecer un gran número de neuronas.

Muy buenos resultados en problemas de reclasificación de imágenes.

Muy buenos resultados en redes convolucionales.

La función de activación ReLU se ha convertido en la más usada en los modelos de aprendizaje profundo durante los últimos años.

IV. Leaky ReLU – Rectified Lineal Unit

La función Leaky ReLU hace una transformación de los valores negativos introducidos en la red neuronal mediante la multiplicación por un coeficiente rectificativo y dejando los valores positivos tales como se entran a la red.

$$(2.9) \quad f(z) = \begin{cases} z & \text{para } z \geq 0 \\ a \cdot z & \text{para } z < 0 \end{cases}$$

Sus características:

Muy parecida a la función ReLU.

penalización de los valores negativos por un coeficiente rectificativo.

No se corta en ningún valor.

Muy buenos resultados en problemas de reclasificación de imágenes.

Muy buenos resultados en redes convolucionales.

V. Softmax – Rectified Lineal Unit

La función softmax se utiliza comúnmente en la capa de salida de una red neuronal para problemas de clasificación multiclase. Toma un vector de valores y lo normaliza para que la suma de todos los valores sea igual a 1. Esto permite interpretar los valores de salida como probabilidades.

$$(2.10) \quad f(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$$

Características:

Se utiliza para representar los resultados de salida de forma probabilística.

Se corta entre los valores entre 0 y 1.

Se puede calcular su derivada fácilmente.

Se permite la normalización de problemas de multiclase.

Buenos resultados en las capas finales.

Ejemplo:

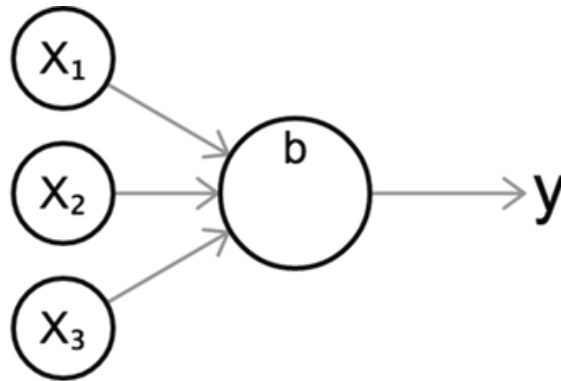


Figura 2.6. Ejemplo de cálculo del valor de función de activación de una neurona.

La **Figura 2.6** muestra una red neuronal compuesta de una capa de entrada de elementos, que produce un valor único en la capa de salida. Como en el caso de función de activación de sigmoidea, primero se calcula la suma de manera ponderada de las entradas, para después se pasa la suma a través de una función de activación (sigmoidea $f(z)$).

$$z = b + w_1x_1 + w_2x_2 + w_3x_3$$

$$y = f(z)$$

2.8. TOPOLOGÍA DE REDES NEURONALES

La topología de una red neuronal se refiere a la estructura y el patrón de conexiones entre las neuronas en la red. Hay varias topologías de redes neuronales que se utilizan en diferentes aplicaciones. En función de la dirección del movimiento de la información se puede realizar una clasificación, la **Figura 2.7** muestra la topología de las redes neuronales existentes.

- **Redes feedforward (conexiones hacia adelante)**

En estas redes, las señales se propagan en una sola dirección, desde la capa de entrada a través de las capas ocultas hasta la capa de salida, sin la existencia de ciclos o retroalimentación. Estas redes son utilizadas en problemas de clasificación, regresión y reconocimiento de patrones.

- **Redes feedback (conexiones hacia atrás o red recurrente)**

Tienen conexiones que forman bucles o circuitos, permitiendo que la información fluya en múltiples direcciones y ciclos. Esto significa que las salidas anteriores de la red pueden retroalimentarse como entradas en pasos de tiempo posteriores. Las redes neuronales recurrentes son adecuadas para problemas que implican secuencias, como el procesamiento del lenguaje natural, el reconocimiento de voz y la predicción de series temporales.

- **Redes neuronales convolucionales**

Estas redes están especialmente diseñadas para el procesamiento de datos en forma de matrices o tensores, como imágenes. Utilizan operaciones de convolución para extraer características relevantes y compartirlas a través de la red. Las redes neuronales convolucionales son ampliamente utilizadas en tareas de visión por computadora, como la clasificación de imágenes y la detección de objetos.

- **Redes neuronales auto asociativas**

Estas redes están diseñadas para aprender y representar relaciones internas en los datos de entrada, permitiendo la reconstrucción o recuperación de los datos originales. Son utilizadas en problemas de compresión de datos, filtrado de ruido y recuperación de información.

- **Redes neuronales de mapas autoorganizados (SOM)**

Estas redes utilizan técnicas de aprendizaje no supervisado para organizar y visualizar datos multidimensionales en una estructura de mapa bidimensional o tridimensional. Son utilizadas en problemas de agrupamiento, clasificación y visualización de datos.

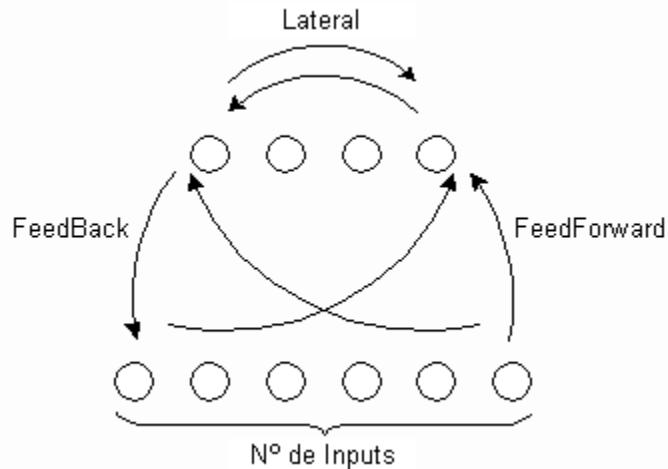


Figura 2.7. Algunas topologías de RNA.

Estas son solo algunas de las topologías de redes neuronales más comunes. Cada topología tiene sus propias características y aplicaciones específicas. La elección de la topología adecuada depende del tipo de problema que se está abordando y los datos disponibles.

2.9. FASE DE ENTRENAMIENTO O APRENDIZAJE DE LA RED NEURONAL

La fase de entrenamiento o aprendizaje de una red neuronal es el proceso en el cual la red se ajusta y se adapta a los datos de entrenamiento para aprender a realizar una tarea específica. Durante esta fase, la red neuronal busca encontrar los pesos y las configuraciones óptimas que minimicen la diferencia entre las salidas esperadas y las salidas producidas por la red.

El entrenamiento de una red neuronal generalmente implica los siguientes pasos:

- ***Inicialización de los pesos***

Se asignan valores iniciales a los pesos de las conexiones entre las neuronas de la red. Esto puede hacerse de manera aleatoria o utilizando técnicas de inicialización específicas.

- ***Propagación hacia adelante (forward propagation)***

Los datos de entrenamiento se introducen en la red neuronal y se propagan a través de las capas de la red, calculando las salidas de cada neurona en función de los pesos actuales y las funciones de activación.

- ***Cálculo de la función de pérdida***

Se calcula una medida de error o pérdida que cuantifica la discrepancia entre las salidas producidas por la red y las salidas esperadas. Esto puede hacerse utilizando diferentes funciones de pérdida, dependiendo del tipo de problema que se esté abordando.

- ***Retropropagación del error (backpropagation)***

El error calculado se propaga hacia atrás a través de la red, ajustando los pesos de las conexiones en función de la contribución de cada peso al error total. Esto se realiza utilizando el gradiente descendente para encontrar la dirección y la magnitud del ajuste necesario para minimizar la función de pérdida.

- ***Actualización de los pesos***

Los pesos de las conexiones se actualizan utilizando un algoritmo de optimización, como el descenso de gradiente, que modifica los pesos en la dirección que reduce el error. Esto se repite iterativamente para múltiples ejemplos de entrenamiento hasta que la red alcance un nivel deseado de precisión o convergencia.

- ***Evaluación del desempeño***

Durante el entrenamiento, se evalúa regularmente el desempeño de la red en un conjunto de datos de validación para monitorear su progreso y evitar el sobreajuste.

Estos pasos se repiten hasta que la red neuronal alcance un nivel de rendimiento satisfactorio o cumpla con los criterios de finalización establecidos. El proceso de entrenamiento puede llevar tiempo y puede requerir ajustes en los hiperparámetros, como la tasa de aprendizaje o el número de épocas, para lograr un buen rendimiento de la red en el problema dado (Torres, 2020).

2.9.1. Ajuste de Hiperparámetros

El ajuste de hiperparámetros (u optimización de hiperparámetros) es el proceso de determinar la combinación correcta de hiperparámetros que maximiza el rendimiento del modelo. Funciona realizando diversas pruebas en un mismo proceso de entrenamiento. Es un proceso importante para encontrar la configuración óptima de los parámetros que afectan el rendimiento y el comportamiento de la red. Los hiperparámetros son variables que no se aprenden durante el entrenamiento de la red, pero que determinan cómo se realiza el entrenamiento y cómo se configura la red. Este proceso, una vez finalizado, le dará el conjunto de valores de hiperparámetros que mejor se adaptan al modelo para obtener resultados óptimos (Theckel, 2016).

Algunos métodos son:

- ***Búsqueda manual o prueba y error***

Se hace uso del conocimiento que tiene sobre el problema, se eligen los parámetros y se observa el resultado. Con base en ese resultado, se puede o no modificar los parámetros. Se repita este proceso varias veces hasta encontrar los parámetros que hacen funcionar muy bien el modelo o hasta se quedan sin tiempo.

- ***Búsqueda de cuadrícula***

Con el conocimiento adquirido sobre el problema se identifican algunos intervalos para elegir los valores de hiperparámetros. Luego se seleccionan un gran número de puntos de esos rangos, se entrena la red utilizando diversas combinaciones de parámetros, se evalúa el desempeño de cada combinación y se selecciona la mejor combinación. Cada vez se podría repetir esa búsqueda de parámetros en un dominio más pequeño con los parámetros que funcionan mejor.

- ***Búsqueda aleatoria***

Es casi igual que la búsqueda en cuadrícula, se hace uso del conocimiento adquirido sobre el problema para poder identificar los intervalos de los valores de los parámetros.

Pero, en vez de escoger valores en los intervalos de una manera secuencial, se hace la selección al azar de manera aleatoria. Se repite este proceso un gran número de veces para encontrar los parámetros que realmente funcionan de manera adecuada o según de lo que ha aprendido el modelo se podría disminuir la búsqueda. Se propone que este sea el método de referencia con el que se deben comparar todos los demás métodos y muestra que tiende a funcionar mejor que los otros métodos (Bergstra et al., 2012).

- ***Optimización bayesiana***

El método consiste en dividir los grandes conjuntos de datos en fragmentos y generar configuraciones de hiperparámetros para los fragmentos utilizando la optimización bayesiana estándar. Se utiliza la información de los fragmentos para el ajuste de hiperparámetros para los grandes conjuntos de datos utilizando una configuración de aprendizaje de transferencia (Theckel et al., 2016).

Algunos términos para el diseño de la red:

- ***Batch Size***

Es el número de ejemplos de entrenamiento que se utilizan en cada actualización de los pesos. Un tamaño de lote grande puede acelerar el entrenamiento, pero también puede requerir más memoria. Un tamaño de lote pequeño puede proporcionar una estimación de gradiente más precisa, pero puede hacer que el entrenamiento sea más lento.

- ***Número de neuronas***

Es el número de nodos que tiene la capa LSTM. Debido a que las redes neuronales tienen una muy baja interpretabilidad, no es posible conocer con certeza hasta qué punto este hiperparámetro influye en una predicción, y debe consumirse un tiempo de computación elevado para probar el resultado con varios valores de este.

- ***Epoch***

Es el número de veces que se recorre el conjunto de entrenamiento completo durante el entrenamiento. En general, en determinado momento, el peso y el residual de cada neurona converge a un cierto valor para una Epoch concreta, y a partir de ahí además de consumir inútilmente tiempo y energía entrenándose, corre el riesgo de producirse sobre entrenamiento (overfitting). Por otro lado, un número bajo de Epoch puede provocar sub-entrenamiento (underfitting) en la red, ya que no aprenda lo suficiente.

- ***Tasa de aprendizaje***

Es la velocidad a la cual los pesos de las conexiones se actualizan durante el entrenamiento. Un valor alto de tasa de aprendizaje puede hacer que el entrenamiento sea más rápido, pero también puede hacer que la red sea inestable o que no converja. Un valor bajo de tasa de aprendizaje puede hacer que el entrenamiento sea más lento, pero más preciso.

2.9.2. Tipo de aprendizaje

En el contexto de las redes neuronales, existen diferentes tipos de aprendizaje que se utilizan para entrenar y ajustar los pesos de la red. Los tipos de aprendizaje más comunes son:

2.9.2.1. Aprendizaje supervisado

En este tipo de aprendizaje, se proporciona a la red neuronal un conjunto de datos de entrenamiento etiquetados, donde cada ejemplo de entrenamiento tiene una entrada y una salida esperada asociada. Durante el entrenamiento, la red ajusta sus pesos para minimizar la diferencia entre las salidas producidas y las salidas esperadas. El aprendizaje supervisado se utiliza para problemas de clasificación y regresión, donde se desea que la red haga predicciones precisas basadas en los datos de entrada. Si el resultado en la neurona de salida no coincide con la deseada, se procederá a la modificación de los pesos de las conexiones, con el objetivo de obtener que la salida obtenida se aproxime a la deseada. Los pesos se van modificando de manera

proporcional al error que se produce entre la salida real de la red y la salida esperada (Alegre, 2003).

Para este tipo de aprendizaje, se puede llevar a cabo de tres maneras:

1) Aprendizaje por corrección de error

Consiste en ajustar los valores de los pesos de las conexiones de la red neuronal en función de la resta entre los valores deseados y los obtenidos a la salida de la red, es decir, en función del error calculado en la salida. Por último, se debe mencionar la regla de aprendizaje de *propagación hacia atrás o de backpropagation*, también conocido como *regla LMS multicapa*, la cual es una generalización de la *regla de aprendizaje Delta* (Rosenblatt, 1958).

2) Aprendizaje por refuerzo

Se reduce a indicar mediante una señal de refuerzo si la salida obtenida en la red se ajusta a la deseada (*éxito = +1 o fracaso = -1*), y en función de ello se ajustan los pesos basándose en un mecanismo de probabilidades (Alegre, 2003).

3) Aprendizaje estocástico

Consiste básicamente en la realización de cambios aleatorios en los valores de los pesos de las conexiones de la red neuronal y evaluar sus resultados a partir del objetivo deseado y de distribuciones probabilísticas. Con los cambios en la red neuronal, es decir, si el comportamiento de la red se cambia o hay un acercamiento al resultado deseado, se podría aceptar el cambio, en el caso contrario, se podría haber una aceptación del cambio en función de umbral o valor preestablecidos.

2.9.2.2. Aprendizaje no supervisado

En el aprendizaje no supervisado, la red neuronal se entrena en un conjunto de datos no etiquetados, sin información sobre las salidas esperadas. El objetivo del aprendizaje no supervisado es descubrir patrones y estructuras ocultas en los datos, como grupos o clusters de datos similares. Este tipo de aprendizaje se utiliza para problemas de

agrupamiento, reducción de dimensionalidad y generación de características. Existen varias posibilidades en cuanto a la interpretación de la salida de estas redes, que dependen de su estructura y del algoritmo de aprendizaje empleado (Hilera, 1995).

Se podría ser de dos tipos tales como:

1) Aprendizaje Hebbiano

Consiste en la modificación de los valores de los pesos con un criterio de correlación entre los valores de las redes neuronales. Esto puede explicarse porque la regla de aprendizaje de Hebb se originó a partir de la neurona biológica clásica, que solamente puede tener dos estados: *activa o inactiva* (Solomatine, 2002).

2) Aprendizaje competitivo y cooperativo

Consiste en la competencia de las neuronas con el objetivo de realizar un trabajo específico. Se permite clasificar los datos de entrada en la primera capa, identificar las clases de los patrones. Si el patrón de entrada se determinó que no pertenece a ninguna de las clases reconocidas anteriormente, entonces la estructura y los pesos de la red neuronal serán ajustados para reconocer la nueva clase (Viñuela, 2004).

2.9.2.3. Aprendizaje por refuerzo

En el aprendizaje por refuerzo, la red neuronal aprende a través de la interacción con un entorno dinámico. La red toma decisiones y realiza acciones en un entorno, y recibe retroalimentación positiva o negativa en forma de recompensas o castigos. El objetivo del aprendizaje por refuerzo es aprender una política óptima que maximice la recompensa acumulada a lo largo del tiempo. Este tipo de aprendizaje se utiliza en problemas de toma de decisiones secuenciales y control de sistemas.

2.10. VALIDACIÓN O FASE DE PRUEBA U OPERACIÓN DE LA RED

La fase de validación o prueba de una red neuronal es la etapa en la que se evalúa el rendimiento y la precisión de la red entrenada en datos que no se utilizaron durante el

entrenamiento. Esta fase es importante para evaluar la capacidad de generalización de la red y verificar si puede hacer predicciones precisas en datos nuevos y no vistos previamente. Durante la fase de validación, se utilizan conjuntos de datos separados, conocidos como conjuntos de validación o conjuntos de prueba, que contienen ejemplos de entrada y las salidas esperadas correspondientes. Estos datos se presentan a la red y se calcula la precisión de las predicciones realizadas por la red en comparación con las salidas esperadas.

La fase de validación puede incluir los siguientes pasos tales como preprocesamiento de datos, propagación hacia adelante (forward propagation), cálculo de la precisión y análisis de resultados. La fase de validación es fundamental para evaluar y garantizar la calidad y el rendimiento de una red neuronal antes de su implementación en aplicaciones del mundo real. Proporciona una estimación objetiva del rendimiento de la red en datos no vistos previamente y ayuda a identificar posibles problemas de sobreajuste o subajuste.

2.10.1. Índices de evaluación del pronóstico

Los índices de evaluación del pronóstico son medidas utilizadas para evaluar y cuantificar el desempeño de un modelo de pronóstico en comparación con los datos reales o esperados. Estos índices proporcionan una forma de medir la precisión y la calidad del pronóstico realizado por el modelo.

Algunas medidas estándar del error son (Dolling et al, 2000):

a) Error Medio Absoluto (Mean Absolute Error)

$$(2.11) \quad MAE = \frac{1}{n} \sum_{i=1}^n |x_i - \hat{x}_i|$$

b) Error Medio Absoluto Porcentual (Mean Absolute Percentage Error)

$$(2.12) \quad MAPE = \frac{100\%}{n} \sum_{i=1}^n \frac{|x_i - \hat{x}_i|}{x_i}$$

c) Raíz Error Medio Cuadrático (Root Mean Squared Error)

$$(2.13) \quad RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2}$$

d) Error Medio Cuadrático Relativo (Mean Squared Relative Error)

$$(2.14) \quad MSRE = \frac{1}{n} \sum_{i=1}^n \frac{(x_i - \hat{x}_i)^2}{x_i^2}$$

e) Coeficiente de Eficiencia (Coefficient of Efficiency)

$$(2.15) \quad CE = 1 - \frac{\sum_{i=1}^n (x_i - \hat{x}_i)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

f) Coeficiente de Determinación (Coefficient of Determination)

$$(2.16) \quad R^2 = \left[\frac{\sum_{i=1}^n (x_i - \bar{x})(\hat{x}_i - \tilde{x})}{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (\hat{x}_i - \tilde{x})^2} \right]^2$$

Donde:

\hat{x}_i : representa valor pronosticado

x_i : representa el valor real de la serie original

\bar{x} : representa valor medio de la serie original

\tilde{x} : valor medio pronosticado

El error medio cuadrático (*Mean Squared Error, MSE*) o equivalentemente *RMSE*, proporciona una buena indicación de la bondad del ajuste en flujos altos, mientras que el error relativo (*MSRE*) proporciona una perspectiva más balanceada de la bondad del ajuste en flujos moderados.

Un valor de *CE* de 0.9 o mayor es una indicación de un pronóstico muy satisfactorio; entre 0.8 y 0.9 representa un modelo aceptablemente bueno y por debajo de 0.8 es considerado no satisfactorio (Shamseldin, 1997).

Estos son solo algunos ejemplos de índices de evaluación del pronóstico, y la elección del índice adecuado depende del problema y del contexto específico. Es importante considerar múltiples índices de evaluación y analizar su significado en conjunto para obtener una evaluación completa y precisa del desempeño del modelo de pronóstico.

2.11. ALGORITMO DE RETROPROPAGACIÓN

El algoritmo de retropropagación es un algoritmo utilizado para entrenar redes neuronales artificiales con múltiples capas. Es una técnica de aprendizaje supervisado que ajusta los pesos de la red en función del error de predicción obtenido durante la fase de entrenamiento. El algoritmo de retropropagación se basa en la propagación hacia atrás de los errores a través de la red.

A continuación, se describen los pasos básicos del algoritmo:

- ***Inicialización de pesos***

Se inicializan aleatoriamente los pesos de todas las conexiones en la red.

- ***Propagación hacia adelante***

Se propaga una muestra de entrenamiento a través de la red, calculando las salidas de cada neurona en función de los pesos actuales y las funciones de activación.

- ***Cálculo del error***

Se calcula el error entre la salida producida por la red y la salida esperada para la muestra de entrenamiento.

- ***Retropropagación del error***

Se retropropaga el error desde la capa de salida hasta la capa de entrada, calculando la contribución de cada neurona al error total de la red. Esto se hace mediante el cálculo del gradiente de error con respecto a los pesos de cada neurona.

- ***Actualización de pesos***

Se actualizan los pesos de todas las conexiones en la red utilizando el gradiente de error calculado. Esto se realiza mediante un algoritmo de optimización, como el descenso del gradiente, que ajusta los pesos en la dirección que minimiza el error.

El algoritmo de retropropagación se utiliza para ajustar los pesos de la red de manera que minimicen el error entre las salidas pronosticadas y las salidas esperadas. A medida que se repiten los pasos del algoritmo para todas las muestras de entrenamiento, la red aprende a ajustar sus pesos de manera que las predicciones se acerquen cada vez más a las salidas esperadas. Es importante destacar que el algoritmo de retropropagación requiere de una función de activación diferenciable para calcular los gradientes de error. Además, la elección adecuada de la arquitectura de la red y los hiperparámetros, como la tasa de aprendizaje, puede influir en la eficiencia y el rendimiento del algoritmo de retropropagación.

2.12. MODELOS ESTOCÁSTICOS

Los modelos estocásticos son modelos matemáticos o estadísticos que incorporan aleatoriedad o incertidumbre en sus componentes o parámetros. Estos modelos son utilizados para describir y analizar fenómenos o procesos que exhiben variabilidad y no pueden ser totalmente deterministas. Los modelos estocásticos se basan en la teoría de la probabilidad y se utilizan en una amplia gama de disciplinas, como estadística, matemáticas, economía, ciencias de la computación, ingeniería, entre otras. Estos modelos permiten modelar y simular situaciones en las que las variables o eventos están sujetos a variaciones aleatorias o inciertas. En el contexto de la actual problemática mundial en torno al agua, existe la creciente necesidad de difundir los conocimientos y las técnicas que permitan lograr los proyectos de aprovechamientos hidráulicos seguros, eficientes y económicos (Escalante et al., 2007).

2.12.1. Modelo AR(p)

El modelo autorregresivo anual AR(p), también conocido como modelo autorregresivo de orden p, es un tipo de modelo estocástico utilizado en el análisis de series de tiempo. En este modelo, el valor de una variable en un momento dado se expresa como una combinación lineal de sus valores anteriores, ponderados por coeficientes autoregresivos.

La forma general del modelo es (Box y Jenkins, 1970):

$$(2.17) \quad Z_t = \sum_{j=1}^p \phi_j Z_{t-j} + \varepsilon_t = \sum_{j=1}^p \phi_j Z_{t-j} + \sigma_\varepsilon \xi_t$$

Donde:

p : orden del modelo auto regresivo.

Z_t : variable estandarizada, con $Z_t = Y_t - \bar{Y}$, $t = 1, 2, \dots, n$, con $E[Y_t] = \mu$ y $\text{Var}(Y_t) = \sigma^2$.

ε_t : serie de residuales que es independiente de Y_t , con $E[\varepsilon_t] = 0$ y $\text{Var}(\varepsilon_t) = \sigma_\varepsilon^2$.

σ_ε : desviación estándar de la serie de residuales ε_t .

ξ_t : serie con distribución Normal estándar con $E[\xi_t] = 0$ y $\text{Var}(\xi_t) = 1$.

Las varianzas σ^2 y σ_ε^2 se relacionan como:

$$(2.18) \quad \sigma_\varepsilon^2 = \sigma^2 [1 - \sum_{j=1}^p \phi_j r_j]$$

ϕ_j : coeficiente auto regresivo.

r_j : coeficiente de auto correlación serial de la variable Z_t .

Distribución normal

Serie normalizada:

Gran parte de las técnicas disponibles para la modelización de las series temporales necesitan que estas series tengan una distribución normal, pero muchas veces no se cumple. Para cubrir este requisito, es recomendable transformar las series registradas X_t a otra Y_t . Con el fin de utilizar los modelos de regresión se requiere establecer un proceso de normalización que puede llevarse a cabo aplicando las siguientes ecuaciones, que son parte del proceso de transformación conocido como de Box-Cox (Chander, 1978).

$$(2.19) \quad Y_t = \frac{(X_t)^\lambda - 1}{\lambda}, \lambda \neq 0, X_t > 0$$

$$(2.20) \quad Y_t = \ln(X_t), \lambda = 0, X_t > 0$$

$X_t, i = 1, 2, \dots, n$: serie para la normalización.

$Y_t, i = 1, 2, \dots, n$: serie transformada en el primer paso.

λ : parámetro de transformación potencial.

Serie estandarizada:

Una vez logrado esto, y de acuerdo con el tipo de serie de que se trate, ya sea anual o periódica, se debe estandarizar la serie como:

$$(2.21) \quad Z_t = Y_t - \bar{Y}, t = 1, 2, \dots, n$$

$$(2.22) \quad Z_t = \frac{Y_t - \bar{Y}}{S_y}, t = 1, 2, \dots, n$$

\bar{Y} y S_y son la media y desviación estándar de la serie normalizada Y_t .

Prueba de independencia de eventos

Para el análisis de frecuencias de las series temporales, se requiere que el conjunto que conforma la muestra X_t^j de la serie j para $i = 1, 2, \dots, n_j$, este compuesta por variables aleatorias. Para probarlo se aplica la prueba de independencia de Anderson (Salas et al., 1988), la cual hace uso del coeficiente de autocorrelación serial r_k^j para diferentes tiempos de retraso k . En caso de que se analiza un registro único, entonces $j = 1$.

Para calcular el coeficiente de autocorrelación serial de retraso k se puede usar la siguiente expresión:

$$(2.23) \quad r_k^j = \frac{\sum_{j=1}^{n_j-k} (X_i^j - \bar{X}^j)(X_{i+k}^j - \bar{X}^j)}{\sum_{i=1}^{n_j} (X_i^j - \bar{X}^j)^2}; \text{ para } r_k^j = 1 \text{ y } k = 1, 2, \dots, \frac{n_j}{3}$$

Donde:

$$(2.24) \quad \bar{X}^j = \sum_{i=1}^{n_j} \frac{X_i^j}{n_j}$$

Para determinar los límites de confianza al 95% para r_k^j se usa la ecuación (2.5).

$$(2.25) \quad r_k^j(95\%) = \frac{-1 \pm 1.96\sqrt{n_j-k-1}}{n_j-k}$$

El correlograma de la muestra es una gráfica, que relaciona los valores de r_k^j como las ordenadas entre los tiempos de retaso k que son las abscisas en correspondencia de los límites de confianza. Si solamente el 10% de los coeficientes de la muestra se encuentra fuera de los límites de confianza entonces la serie X_i^j es independiente, los datos son aleatorios.

- Función auto covarianza de la serie normalizada:

$$(2.26) \quad C_k = \frac{1}{N} \sum_{t=1}^{n-k} (Y_t - \bar{Y})(Y_{t+k} - \bar{Y}), \text{ Para } 0 < k < N/3$$

- Los coeficientes de auto correlación serial se obtiene como:

$$(2.27) \quad r_k = \frac{C_k}{C_0}$$

Número de Box-Müller

La serie ξ_t se obtiene a través del procedimiento propuesto por (Box y Müller 1958), el cual consiste en proponer números u_1 y u_2 con distribución uniforme en el intervalo (0,1). Este procedimiento permite calcular números aleatorios mediante la distribución normal estándar por parejas y es necesario determinar los valores tantas veces como el tamaño de la muestra requerido con la finalidad de generar las muestras sintéticas.

$$(2.28) \quad \xi_1 = [2\ln(1/u_1)]^{1/2} \cos(2\pi u_2)$$

$$(2.29) \quad \xi_2 = [2\ln(1/u_1)]^{1/2} \sin(2\pi u_2)$$

Serie de residuales

Una vez que se han obtenido los parámetros del modelo y se debe verificar que la serie de residuales tiene una distribución Normal y es independiente.

$$(2.30) \quad Z_t = \sum_{j=1}^p \phi_j Z_{t-j} + \varepsilon_t, t = 1, 2, \dots, n$$

La normalidad se verifica considerando que el coeficiente de asimetría de los residuales es aproximadamente igual a cero. La independencia se comprueba a través de la prueba de Anderson.

Criterio de información Akaike

Para seleccionar el mejor modelo que describe el comportamiento de la serie de tiempo se utiliza el criterio de información de Akaike (Akaike, 1974).

$$(2.31) \quad AIC = N[\ln\delta_\varepsilon^2] + 2(p + q)$$

El mejor modelo según los valores de AIC es aquel que tiene el mínimo valor de AIC.

Criterio de Información Bayesiano

También para la selección del mejor modelo se puede utilizar este criterio de selección. Se prefiere el modelo que tenga el mínimo valor del estadístico.

$$(2.32) \quad BIC = N[\ln\delta_\varepsilon^2] + (p + q)\ln(N)$$

El *BIC* penaliza los parámetros adicionales más severamente que el *AIC*, conduciendo a modelos más simples.

Modelo AR (1)

La forma del modelo es:

$$(2.33) \quad Z_t = \phi_1 Z_{t-1} + \varepsilon_t$$

$$(2.34) \quad Z_t = \phi_1 Z_{t-1} + \sigma_\varepsilon \xi_t$$

El parámetro autorregresivo se estima como:

$$(2.35) \quad \phi_1 = r_1$$

Para cubrir la condición de estacionariedad se debe cumplir que:

$$\begin{cases} -1 < \phi_1 < 1 \\ -1 < r_1 < 1 \end{cases}$$

La serie de residuales se obtiene para $t = 2, 3, \dots, n$ y se considera que $\varepsilon_1 = 0$

$$(2.36) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1}$$

La varianza de los residuales se obtiene a partir de la expresión como:

$$(2.37) \quad \sigma_\varepsilon^2 = \sigma^2(1 - \phi_1^2)$$

σ^2 : varianza de la serie normalizada de Y_t

Se calcula la varianza insesgada mediante la ecuación (2.38):

$$(2.38) \quad \sigma_\varepsilon^2 = \frac{N}{N-1} \sigma^2(1 - \phi_1^2)$$

Modelo AR (2)

La forma del modelo es:

$$(2.39) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \varepsilon_t$$

$$(2.40) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \sigma_\varepsilon \xi_t$$

Los parámetros autorregresivos se estiman como:

$$(2.41) \quad \phi_1 = \frac{r_1(1-r_2)}{1-r_1^2}$$

$$(2.42) \quad \phi_2 = \frac{r_2-r_1^2}{1-r_1^2}$$

Para cubrir la condición de estacionariedad se debe cumplir que:

$$\left\{ \begin{array}{l} \phi_1 + \phi_2 < 1 \\ \phi_2 - \phi_1 < 1 \\ -1 < \phi_2 < 1 \\ -1 < r_1 < 1 \\ -1 < r_2 < 1 \\ r_1^2 < \frac{(1+r_2)}{2} \end{array} \right.$$

La serie de residuales se obtiene para $t = 3, 4, \dots, n$ y se considera que $\varepsilon_1 = 0$ y $\varepsilon_2 = 0$

$$(2.43) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2}$$

La varianza de los residuales se obtiene a partir de la expresión como:

$$(2.44) \quad \sigma_\varepsilon^2 = \sigma^2 \frac{(1+\phi_2)}{(1-\phi_2)} [(1-\phi_2)^2 - \phi_1^2]$$

σ^2 : es la varianza de la serie normalizada Y_t

La versión insesgada de la expresión anterior se obtiene por:

$$(2.45) \quad \sigma_\varepsilon^2 = \sigma^2 \frac{N}{N-2} \frac{(1+\phi_2)}{(1-\phi_2)} [(1-\phi_2)^2 - \phi_1^2]$$

Modelo AR (3)

La forma del modelo es:

$$(2.46) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \varepsilon_t$$

$$(2.47) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \sigma_\varepsilon \xi_t$$

Los parámetros auto regresivos se estiman como:

$$(2.48) \quad \phi_1 = \frac{(1-r_1^2)(r_1-r_3) - (1-r_2)(r_1 r_2 - r_3)}{(1-r_2)(1-2r_1^2+r_2)}$$

$$(2.49) \quad \phi_2 = \frac{(1-r_2)(r_2+r_2^2-r_1^2-r_1 r_3)}{(1-r_2)(1-2r_1^2+r_2)}$$

$$(2.50) \quad \phi_3 = \frac{(r_1-r_3)(r_1^2-r_2) - (1-r_2)(r_1 r_2 - r_3)}{(1-r_2)(1-2r_1^2+r_2)}$$

La serie de residuales se obtiene para $t = 4, 5, \dots, n$ y se considera que $\varepsilon_1 = 0$, $\varepsilon_2 = 0$ y $\varepsilon_3 = 0$

$$(2.51) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2} - \phi_3 Z_{t-3}$$

La varianza de los residuales se obtiene a partir de la expresión como:

$$(2.52) \quad \hat{\sigma}_\varepsilon^2 = \sigma^2 [1 - \sum_{j=1}^p \hat{\phi}_j \hat{r}_j]$$

σ^2 : es la varianza de la serie normalizada Y_t

2.12.2. Modelo ARMA (p, q)

En estadística, los modelos autorregresivos de media móvil (en inglés Autorregresiva Moving Average models, abreviados ARMA), también llamados Modelos Box-Jenkins, se aplican a series temporales de datos (Escalante et al., 2002). Los modelos ARMA contienen dos partes principales la parte autorregresiva (AR) y la otra parte media móvil (MA).

La forma del modelo es:

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \dots - \theta_q \varepsilon_{t-q}$$

$$(2.53) \quad Z_t = \sum_{j=1}^p \phi_j Z_{t-j} + \varepsilon_t - \sum_{j=1}^q \theta_j \varepsilon_{t-j}$$

$$Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2} - \dots - \theta_q \sigma_\varepsilon \xi_{t-q}$$

$$(2.54) \quad Z_t = \sum_{j=1}^p \phi_j Z_{t-j} + \sigma_\varepsilon \xi_t - \sum_{j=1}^q \theta_j \sigma_\varepsilon \xi_{t-j}$$

Estimación de parámetros:

$$(2.55) \quad \sigma_\varepsilon^2 = \frac{C_0}{1 + \theta_1^2 + \dots + \theta_q^2}$$

$$(2.56) \quad \theta_j = -\left(\frac{C_j}{\sigma_\varepsilon^2} - \theta_1 \theta_{j+1} - \theta_2 \theta_{j+2} - \dots - \theta_q \theta_{j+q}\right)$$

Auto covarianza (se utiliza la serie de residuales):

$$(2.57) \quad C_k = \frac{1}{N} \sum_{t=1}^{n-k} (Z_t - \bar{Z})(Z_{t+k} - \bar{Z})$$

Proceso de medias móviles MA (1):

$$(2.58) \quad \sigma_\varepsilon^2 = \frac{C_0}{1 + \theta_1^2}$$

$$(2.59) \quad \theta_1 = -\frac{C_1}{\sigma_\varepsilon^2}$$

Donde C_0 y C_1 son la auto covarianza de la serie de residuales.

Proceso de medias móviles MA (2):

$$(2.60) \quad \sigma_{\varepsilon}^2 = \frac{C_0}{1+\theta_1^2+\theta_2^2}$$

$$(2.61) \quad \theta_1 = -\frac{C_2}{\sigma_{\varepsilon}^2}$$

$$(2.62) \quad \theta_2 = -\left(\frac{C_1}{\sigma_{\varepsilon}^2} - \theta_1\theta_2\right)$$

Donde C_0 , C_1 y C_2 son la auto covarianza de la serie de residuales.

Modelo ARMA (1,1)

La forma del modelo es:

$$(2.63) \quad Z_t = \phi_1 Z_{t-1} + \varepsilon_t - \theta_1 \varepsilon_{t-1}$$

$$(2.64) \quad Z_t = \phi_1 Z_{t-1} + \sigma_{\varepsilon} \xi_t - \theta_1 \sigma_{\varepsilon} \xi_{t-1}$$

Proceso MA (1):

$$(2.65) \quad \sigma_{\varepsilon}^2 = \frac{C_0}{1+\theta_1^2}$$

$$(2.66) \quad \theta_1 = -\frac{C_1}{\sigma_{\varepsilon}^2}$$

Residuales del modelo ARMA (1,1):

$$(2.67) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} + \theta_1 \varepsilon_{t-1}$$

Generación de muestras sintéticas con el modelo ARMA (1,1):

$$(2.68) \quad Z_t = \phi_1 Z_{t-1} + \sigma_{\varepsilon} \xi_t - \theta_1 \sigma_{\varepsilon} \xi_{t-1}$$

Modelo ARMA (1,2)

La forma del modelo es:

$$(2.69) \quad Z_t = \phi_1 Z_{t-1} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2}$$

$$(2.70) \quad Z_t = \phi_1 Z_{t-1} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2}$$

Proceso MA (2):

$$(2.71) \quad \sigma_\varepsilon^2 = \frac{C_0}{1+\theta_1^2+\theta_2^2}$$

$$(2.72) \quad \theta_1 = -\frac{C_2}{\sigma_\varepsilon^2}$$

$$(2.73) \quad \theta_2 = -\left(\frac{C_1}{\sigma_\varepsilon^2} - \theta_1 \theta_2\right)$$

Residuales del modelo ARMA (1,2):

$$(2.74) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2}$$

Generación de muestras sintéticas con el modelo ARMA (1,2):

$$(2.75) \quad Z_t = \phi_1 Z_{t-1} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2}$$

Modelo ARMA (2,1)

La forma del modelo es:

$$(2.76) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \varepsilon_t - \theta_1 \varepsilon_{t-1}$$

$$(2.77) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1}$$

Proceso MA (1):

$$(2.78) \quad \sigma_\varepsilon^2 = \frac{C_0}{1+\theta_1^2}$$

$$(2.79) \quad \theta_1 = -\frac{C_1}{\sigma_\varepsilon^2}$$

Residuales del modelo ARMA (2,1):

$$(2.80) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2} + \theta_1 \varepsilon_{t-1}$$

Generación de muestras sintéticas con el modelo ARMA (2,1):

$$(2.81) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1}$$

Modelo ARMA (2, 2)

La forma del modelo es:

$$(2.82) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2}$$

$$(2.83) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2}$$

Proceso MA (2):

$$(2.84) \quad \sigma_\varepsilon^2 = \frac{C_0}{1 + \theta_1^2 + \theta_2^2}$$

$$(2.85) \quad \theta_1 = -\frac{C_2}{\sigma_\varepsilon^2}$$

$$(2.86) \quad \theta_2 = -\left(\frac{C_1}{\sigma_\varepsilon^2} - \theta_1 \theta_2\right)$$

Residuales del modelo ARMA (2, 2):

$$(2.87) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2}$$

Generación de muestras sintéticas con el modelo ARMA (2, 2):

$$(2.88) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2}$$

Modelo ARMA (3,1)

La forma del modelo es:

$$(2.89) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \varepsilon_t - \theta_1 \varepsilon_{t-1}$$

$$(2.90) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1}$$

Proceso MA (1):

$$(2.91) \quad \sigma_\varepsilon^2 = \frac{C_0}{1 + \theta_1^2}$$

$$(2.92) \quad \theta_1 = -\frac{C_1}{\sigma_\varepsilon^2}$$

Residuales del modelo ARMA (3,1):

$$(2.93) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2} - \phi_3 Z_{t-3} + \theta_1 \varepsilon_{t-1}$$

Generación de muestras sintéticas con el modelo ARMA (3,1):

$$(2.94) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1}$$

Modelo ARMA (3,2)

La forma del modelo es:

$$(2.95) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2}$$

$$(2.96) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2}$$

Proceso MA (2):

$$(2.97) \quad \sigma_\varepsilon^2 = \frac{c_0}{1 + \theta_1^2 + \theta_2^2}$$

$$(2.98) \quad \theta_1 = -\frac{c_2}{\sigma_\varepsilon^2}$$

$$(2.99) \quad \theta_2 = -\left(\frac{c_1}{\sigma_\varepsilon^2} - \theta_1 \theta_2\right)$$

Residuales del modelo ARMA (3,2):

$$(2.100) \quad \varepsilon_t = Z_t - \phi_1 Z_{t-1} - \phi_2 Z_{t-2} - \phi_3 Z_{t-3} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2}$$

Generación de muestras sintéticas con el modelo ARMA (3,2):

$$(2.101) \quad Z_t = \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \phi_3 Z_{t-3} + \sigma_\varepsilon \xi_t - \theta_1 \sigma_\varepsilon \xi_{t-1} - \theta_2 \sigma_\varepsilon \xi_{t-2}$$

2.12.3. Modelos de pronóstico

Los modelos estocásticos permiten la generación de muestras sintéticas con propósitos de planeación de aprovechamientos hidráulicos, sin embargo, también son una

herramienta para el pronóstico de eventos con propósitos de operación. Presentar y aplicar el modelo general de pronóstico estocástico (p, q).

El modelo tiene la forma:

$$(2.102) \quad Z_t(L) = \hat{\varphi}_1 Z_t + \dots + \hat{\varphi}_p Z_{t-p} - \hat{\theta}_1 \varepsilon_t - \dots - \hat{\theta}_q \varepsilon_{t-q+1}$$

$Z_t(L)$: es el valor pronosticado del paso L

Z_t : es el valor de la serie estandarizada en el tiempo t .

$\hat{\varphi}_1 + \dots + \hat{\varphi}_p$: son los parámetros autorregresivos

$\hat{\theta}_1 - \dots - \hat{\theta}_q$: son los parámetros de promedios móviles

$\varepsilon_t - \dots - \varepsilon_{t-q+1}$: son los residuales del modelo ARMA (p, q)

Los modelos particulares son:

$$(2.103) \quad AR(1): Z_t(L) = \hat{\varphi}_1 Z_t$$

$$(2.104) \quad AR(2): Z_t(L) = \hat{\varphi}_1 Z_t + \hat{\varphi}_2 Z_{t-2+1}$$

$$(2.105) \quad AR(3): Z_t(L) = \hat{\varphi}_1 Z_t + \hat{\varphi}_2 Z_{t-2+1} + \hat{\varphi}_3 Z_{t-3+1}$$

$$(2.106) \quad ARMA(1,1): Z_t(L) = \hat{\varphi}_1 Z_t - \hat{\theta}_1 \varepsilon_t$$

$$(2.107) \quad ARMA(1,2): Z_t(L) = \hat{\varphi}_1 Z_t - \hat{\theta}_1 \varepsilon_t - \hat{\theta}_2 \varepsilon_{t-2+1}$$

$$(2.108) \quad ARMA(2,1): Z_t(L) = \hat{\varphi}_1 Z_t + \hat{\varphi}_2 Z_{t-2+1} - \hat{\theta}_1 \varepsilon_t$$

$$(2.109) \quad ARMA(2,2): Z_t(L) = \hat{\varphi}_1 Z_t + \hat{\varphi}_2 Z_{t-2+1} - \hat{\theta}_1 \varepsilon_t - \hat{\theta}_2 \varepsilon_{t-2+1}$$

$$(2.110) \quad ARMA(3,1): Z_t(L) = \hat{\varphi}_1 Z_t + \hat{\varphi}_2 Z_{t-2+1} + \hat{\varphi}_3 Z_{t-3+1} - \hat{\theta}_1 \varepsilon_t$$

$$(2.111) \quad ARMA(3,2): Z_t(L) = \hat{\varphi}_1 Z_t + \hat{\varphi}_2 Z_{t-2+1} + \hat{\varphi}_3 Z_{t-3+1} - \hat{\theta}_1 \varepsilon_t - \hat{\theta}_2 \varepsilon_{t-2+1}$$

En un AR (1) siempre $Z_t(L) < Z_t$ en tanto que en un ARMA (1,1) $Z_t(L) < Z_t$ o $Z_t(L) > Z_t$.

Cálculo del error medio cuadrático (EMC):

$$(2.112) \quad EMC = \sqrt{\frac{1}{N} \sum_{i=1}^N (X_t(L) - X_t)^2}$$

$X_t(L)$: valor pronosticado

X_t : valor real

N : número de datos de la muestra

2.12.4. Modelo ARIMA (p, d, q)

Box y Jenkins han desarrollado los modelos Autorregresivo de Medias Móviles ARIMA(p,d,q) para modelar procesos estocásticos no estacionarios. El modelo ARIMA(p,d,q) (AutoRegresive Integrated Moving Average en inglés), está dividido en tres componentes AR (Autorregresivo), I(Integrado) y MA (Medias Móviles), en los modelos ARIMA(p,d,q), p representa el orden del proceso auto regresivo, d número de diferencias que son necesarias para que el proceso sea estacionario y q representa el orden del proceso de medias móviles (Box y Jenkins, 1970).

La forma del modelo es:

$$(2.113) \quad Z_t = c + \sum_{j=1}^p \phi_j Z_{t-j} + \varepsilon_t + \sum_{j=1}^q \theta_j \varepsilon_{t-j}$$

$$Z_t = c + \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \dots + \phi_p Z_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q}$$

Proceso estocástico

Un proceso estocástico es un modelo matemático que describe la evolución de una variable a lo largo del tiempo, donde el valor de la variable en cada instante de tiempo es aleatorio o incierto. También se le conoce como proceso aleatorio. Los procesos estocásticos se utilizan en diversas disciplinas, como la estadística, la economía, la física, la ingeniería y la ciencia de datos. Son herramientas fundamentales para modelar y comprender fenómenos complejos y estocásticos que ocurren en la naturaleza y en el mundo real.

Ruido blanco

El ruido blanco es un tipo de señal o proceso estocásticos en el que las variables aleatorias que lo componen son independientes e idénticamente distribuidas con una distribución de probabilidad constante. En otras palabras, el ruido blanco no tiene correlación temporal y cada muestra de la señal es independiente y se distribuye de manera uniforme.

Es importante tener en cuenta que el ruido blanco idealizado puede ser difícil de encontrar en la práctica, ya que la mayoría de las señales reales están sujetas a diferentes fuentes de ruido y tienen cierto grado de correlación temporal. Sin embargo, el ruido blanco se utiliza como una simplificación o modelo teórico para describir ciertos fenómenos y como una referencia para comparar el rendimiento de otros procesos o sistemas.

Prueba Ljung-Box

Para validar el ruido blanco en una serie temporal se usa la hipótesis de la prueba de Ljung-Box. En la prueba de Ljung-Box si el valor de $p > 0.05$ (5% nivel de significancia), significa que el modelo se ajusta bien y existe ruido blanco. Sin embargo, si el valor de $p < 0.05$, significa que el modelo no se ajusta bien y por lo tanto no existe ruido blanco.

Sendero aleatorio

Un sendero aleatorio, también conocido como caminata aleatoria, es un tipo de proceso estocástico en el que una variable cambia de manera aleatoria a medida que avanza el tiempo. Cada cambio o paso en el sendero aleatorio se toma de forma independiente y con una distribución de probabilidad especificada.

Es importante destacar que un sendero aleatorio es un concepto matemático y puede diferir de las caminatas o trayectorias reales en el mundo físico. Sin embargo, el sendero aleatorio proporciona una representación simplificada y útil para estudiar y analizar procesos estocásticos.

Estacionariedad

La estacionariedad es una propiedad importante en el análisis de series de tiempo y se refiere a la constancia o invariancia de ciertas características estadísticas a lo largo del tiempo. Una serie de tiempo se considera estacionaria si sus propiedades estadísticas no cambian con el tiempo. La estacionariedad de una serie temporal significa que la serie no tiene una tendencia a largo plazo, es decir, es estable alrededor del mismo promedio. En caso contrario, se dice que la serie temporal es no estacionaria.

Existen dos tipos principales de estacionariedad:

- ***Estacionariedad estricta***

También conocida como estacionariedad estricta o estacionariedad en sentido estricto, implica que todas las características estadísticas de la serie de tiempo son constantes a lo largo del tiempo. Esto incluye la media, la varianza y las autocovarianzas. En otras palabras, la distribución de probabilidad de los valores en la serie de tiempo no cambia con el tiempo.

- ***Estacionariedad en sentido amplio***

También conocida como estacionariedad en sentido amplio o estacionariedad débil, se refiere a que la media y la autocovarianza de la serie de tiempo son constantes a lo largo del tiempo, pero la varianza puede variar. Esto significa que la serie de tiempo puede tener una media constante y una estructura de dependencia temporal constante, pero la dispersión de los valores puede cambiar a lo largo del tiempo.

La estacionariedad es una propiedad deseable en el análisis de series de tiempo, ya que facilita la modelización y el pronóstico de la serie. Cuando una serie de tiempo es estacionaria, se pueden aplicar métodos estadísticos más sencillos y las propiedades del proceso subyacente se mantienen constantes.

Prueba de Dickey-Fuller Aumentada (ADF)

La prueba de Dickey-Fuller Aumentada es una prueba estadística utilizada para determinar si una serie de tiempo es estacionaria o no estacionaria. Esta prueba es una extensión de la prueba de Dickey-Fuller original, que se utiliza para analizar la

presencia de raíces unitarias en una serie de tiempo. La prueba de Dickey-Fuller Aumentada se basa en un modelo autorregresivo (AR) y se utiliza para contrastar la hipótesis nula de que la serie de tiempo tiene una raíz unitaria (es decir, es no estacionaria) frente a la hipótesis alternativa de que la serie de tiempo es estacionaria.

La estacionalidad, así como la estacionariedad, también puede detectarse a través de las funciones de auto correlación (ACF) y auto correlación parcial (ACFP) estimadas por la prueba Augmentada de Dickey-Fuller. Una auto correlación positiva significa que un valor alto ahora, probablemente dará un valor alto en el próximo periodo.

Si la serie temporal presenta una tendencia, lo primero que debe de hacerse es convertirla en estacionaria mediante una diferenciación de orden d . Una vez diferenciada la serie, una buena estrategia consiste en comparar los correlogramas de la función de auto correlación (ACF) y la función de auto correlación parcial (ACFP). Puede ser conveniente utilizar logaritmos de la serie o aplicar la transformación de Box-Cox.

$$(2.114) \quad \Delta Z_t = \alpha + \beta t + \gamma Z_{t-1} + \delta_1 \Delta Z_{t-1} + \delta_2 \Delta Z_{t-2} + \dots$$

La prueba de Dickey-Fuller Aumentada considera diferentes modelos, como el modelo sin tendencia, el modelo con tendencia lineal y el modelo con tendencia lineal y término de error autorregresivo. Estos modelos permiten tener en cuenta diferentes características de la serie de tiempo y evaluar su estacionariedad. La prueba de Dickey-Fuller Aumentada es ampliamente utilizada en el análisis de series de tiempo y es una herramienta importante para determinar la estacionariedad de una serie y tomar decisiones adecuadas en el modelado y pronóstico de datos temporales.

Donde Δ es el operador de la primera diferencia. La hipótesis nula de la prueba (ADF) es que la serie de tiempo no es estacionaria.

$$ADF = \frac{\gamma}{SE(\gamma)}$$

Una vez que el valor del estadístico de prueba p se calcula, debe ser comparado con el valor crítico relevante para la prueba de Dickey-Fuller, es decir, $p < 0.05$ (5% nivel de significancia de los datos). Si $p > 0.05$ se itera otra vez los datos calculando la diferencia, hasta que se rechace la hipótesis nula y que los datos sean estacionarios.

2.13. CARACTERÍSTICAS ESTADÍSTICAS

2.13.1. Medidas de tendencia central

Las medidas de tendencia central son aquellos números que definen el valor alrededor del cual se concentra el conjunto de observaciones.

- **Media**

Si x_1, x_2, \dots, x_n representa el conjunto de datos $\{x_i\}$ para $i = 1, 2, \dots, n$ elementos, el valor de la media o promedio aritméticos se obtiene como:

$$(2.115) \quad \bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} = \frac{1}{n} \sum_{i=1}^n x_i$$

- **Moda**

Es el valor que repita con un gran número de veces en una muestra. Puede ser que no exista, pero puede existir más de una moda "Mo". Valor que se representa con la mayor frecuencia.

- **Mediana**

La mediana \tilde{x} es el valor de la serie que se encuentra en la posición central, cuando los datos se ordenan de menor a mayor magnitud. El valor se determina considerando si la serie tiene un tamaño de muestra n impar o par. Valor de una serie que ocupa la posición central.

$$(2.116) \quad \tilde{x} = x_{\left(\frac{n+1}{2}\right)}, \text{ Sí } n \text{ es impar}$$

$$(2.117) \quad \tilde{x} = \frac{1}{2} \left[x_{\left(\frac{n}{2}\right)} + x_{\left(\frac{n}{2} + 1\right)} \right], \text{ Sí } n \text{ es par}$$

2.13.2. Medidas de dispersión

Son números que proveen información adicional acerca del grado de variabilidad de los datos.

- **Rango muestral**

Para una muestra de datos que se ordena de menor a mayor, el rango es la diferencia entre el mayor valor $x_{(n)}$ y menor valor $x_{(1)}$.

$$(2.118) \quad R = x_{(n)} - x_{(1)}$$

- **Varianza muestral**

Esta medida se basa en la cuantificación de las distancias de los datos x_i con respecto al valor de la media \bar{x} . Se obtiene a través de la expresión:

$$(2.119) \quad S^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

O también:

$$(2.120) \quad S^2 = \frac{n \sum_{i=1}^n x_i^2 - \sum_{i=1}^n x_i^2}{n(n-1)}$$

- **Desviación estándar**

Es la raíz cuadrada de la varianza muestral de la serie.

$$(2.121) \quad S = \sqrt{S^2}$$

- **Coficiente de variación**

S es la desviación estándar o típica de la serie. \bar{x} es la media de la serie en valor absoluto y diferente de cero.

$$(2.122) \quad C_V = \frac{S}{\bar{x}}$$

2.13.3. Medidas de posición

Son números que dividen al grupo de datos ordenados, en grupos de aproximadamente igual cantidad de datos con el propósito de resaltar su ubicación.

- **Cuartiles de la muestra**

Son números que dividen al grupo de datos ordenados de menor a mayor, en grupos de aproximadamente 25% de los datos.

Primer cuartil (Q_1)

A la izquierda el 25% de los datos y a la derecha el 75% de los datos.

Segundo cuartil (Q_2) (mediana)

A la izquierda el 50% de los datos y a la derecha el 50% de los datos.

Tercer cuartil (Q_3)

A la vez que lee el 75% los datos y a la derecha el 25% de los datos.

- **Deciles de una muestra**

El decil es una medida estadística que se utiliza para indicar el valor por debajo del cual se encuentra un determinado porcentaje de observaciones y son números que dividen al grupo de datos ordenados de menor x_1 a mayor x_n , en grupos aproximadamente de 10% de los datos. Los deciles se obtienen como:

$$(2.123) \quad D(m) = m \left(\frac{n}{10} \right)$$

| | | | | | |
|------------------|------------------|------------------|----|----|------------------|
| Decil m=1 | Decil m=2 | Decil m=3 | .. | .. | Decil m=9 |
| n/10 | 2n/10 | 3n/10 | | | 9n/10 |

Si el cálculo de la posición de un decil no coincide con uno de los valores de los datos, entonces queda entre los dos valores a y b , y se procede a hacer una interpolación entre los dos valores.

- **Percentiles de una muestra**

El percentil $P(i)$ es una medida estadística de posición que indica, una vez ordenados de menor x_1 a mayor x_n , el valor de la variable por debajo del cual se encuentra un porcentaje dado i de observaciones en un grupo. Por ejemplo, el percentil $i = 10$, es el valor bajo el cual se encuentra el 10% de las observaciones. El percentil $P(50)$ como la mediana de la muestra.

$$(2.124) \quad \text{Índice} = \frac{n \cdot i}{100}$$

Donde n es el tamaño de la muestra, i el percentil e índice es un número real con una parte entera E y una parte decimal D . Entonces se obtiene el valor que define el percentil buscado como:

$$P(i) = x_{(E+1)}, \text{ cuando } D \neq 0$$

$$P(i) = \frac{x_{(E)} + x_{(E+1)}}{2}, \text{ cuando } D = 0$$

2.13.4. Medidas de distribución

Las medidas de distribución se utilizan para describir y caracterizar la distribución de datos en un conjunto de observaciones. Dependiendo del contexto y la naturaleza de los datos, es posible que se utilicen otras medidas adicionales para caracterizar la distribución de los datos, como la asimetría, la curtosis, los percentiles truncados, entre otros.

- **Coefficiente de Curtosis K**

El valor de referencia o desviación respecto a la Normal donde su Curtosis es 3 es:

$$(2.125) \quad k = \left[\frac{n(n+1)}{(n-1)(n-2)(n-3)} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s} \right)^4 \right] - \frac{3(n-1)^2}{(n-2)(n-3)}$$

Coefficiente de Curtosis de la muestra:

$$(2.126) \quad K = 3 + k$$

CAPÍTULO 3:

3. METODOLOGÍA

Este capítulo se refiere al análisis de series temporales aplicando modelos autorregresivos AR(p), autorregresivos de medios móviles ARMA(p,q), autorregresivos integrados de medios móviles ARIMA(p,d,q) y redes neuronales artificiales recurrentes (LSTM) utilizando el conjunto de datos de estaciones hidrométricas que se encuentran aguas arriba de la presa La Angostura. Como resultado de este capítulo se obtendrá el modelo de pronóstico que mejor se adapte a las condiciones de las series de tiempo para el pronóstico de caudal medio diario.

3.1. MEDIO FÍSICO

3.1.1. Zona de estudio

La zona de estudio se encuentra aguas arriba de la Presa La Angostura (**Figura 3.2** y **Figura 3.3**), conocida oficialmente como Central Hidroeléctrica Dr. Belisario Domínguez (Presa La Angostura), es la presa que tiene el embalse más grande del país. Se encuentra ubicada en la zona más alta del cauce del Río Grijalva en territorio tabasqueño, en el municipio La Concordia, Chiapas. Inició operaciones el 14 de julio de 1976.

La central hidroeléctrica tiene una potencia instalada de 900 mega watts (MW) para la generación de energía eléctrica (**Tabla 3.1**), con un embalse que tiene una capacidad ordinaria aproximada de 10 727 hm³ y una extraordinaria de hasta 12 762 hm³, así siendo la mayor de México. La altura de su cortina es de 143 metros (CONAGUA, 2010).

Las presas tales como La Angostura, Chicoasén, Malpaso y Peñitas forman el sistema hidroeléctrico más importante de México, con su disposición en cascada llegan a alcanzar una capacidad total de 4800 MW que pueden generar hasta 19 344 GWH

(Giga Watts por Hora) eso representa hasta el 44% de la energía hidroeléctrica total del país.

Tabla 3.1. Presas del sistema Grijalva con capacidad instalada y generación media anual (INEGI, 2014).

| Nombre de la Presa | Capacidad Instalada (MW) | Generación media anual (GWH) | Porcentaje de capacidad hidroeléctrica del país (%) |
|------------------------------------|--------------------------|------------------------------|---|
| Belisario Domínguez (La Angostura) | 900 | 3991 | 9.08% |
| Manuel Moreno Torres (Chicoasén) | 2400 | 8080 | 18.37% |
| Nezahualcóyotl (Malpaso) | 1080 | 5052 | 11.50% |
| Ángel Albino Corzo (Peñitas) | 420 | 2221 | 5.05% |
| Total | 4800 | 19344 | 44.00% |

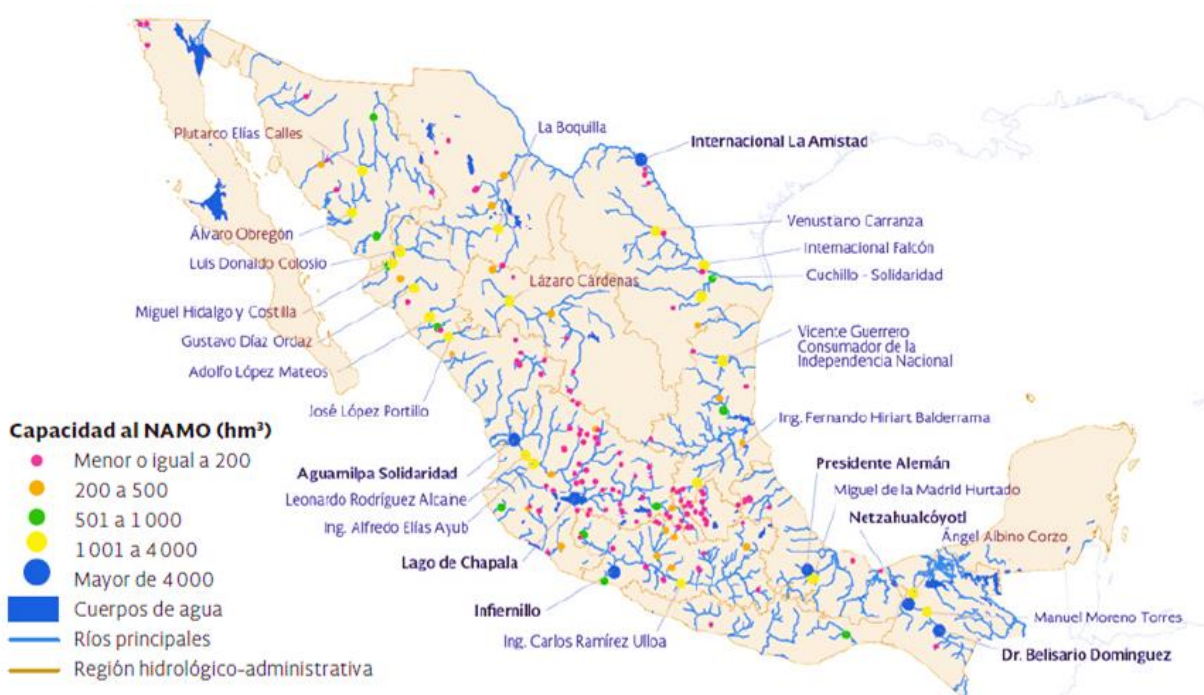


Figura 3.1. Principales presas en México (Conagua, 2018)⁵.

⁵ **Nota:** Se muestran los nombres de las presas con capacidad mayor a 1 000 hm³. En negritas se destacan las de capacidad mayor a 4 000 hm³

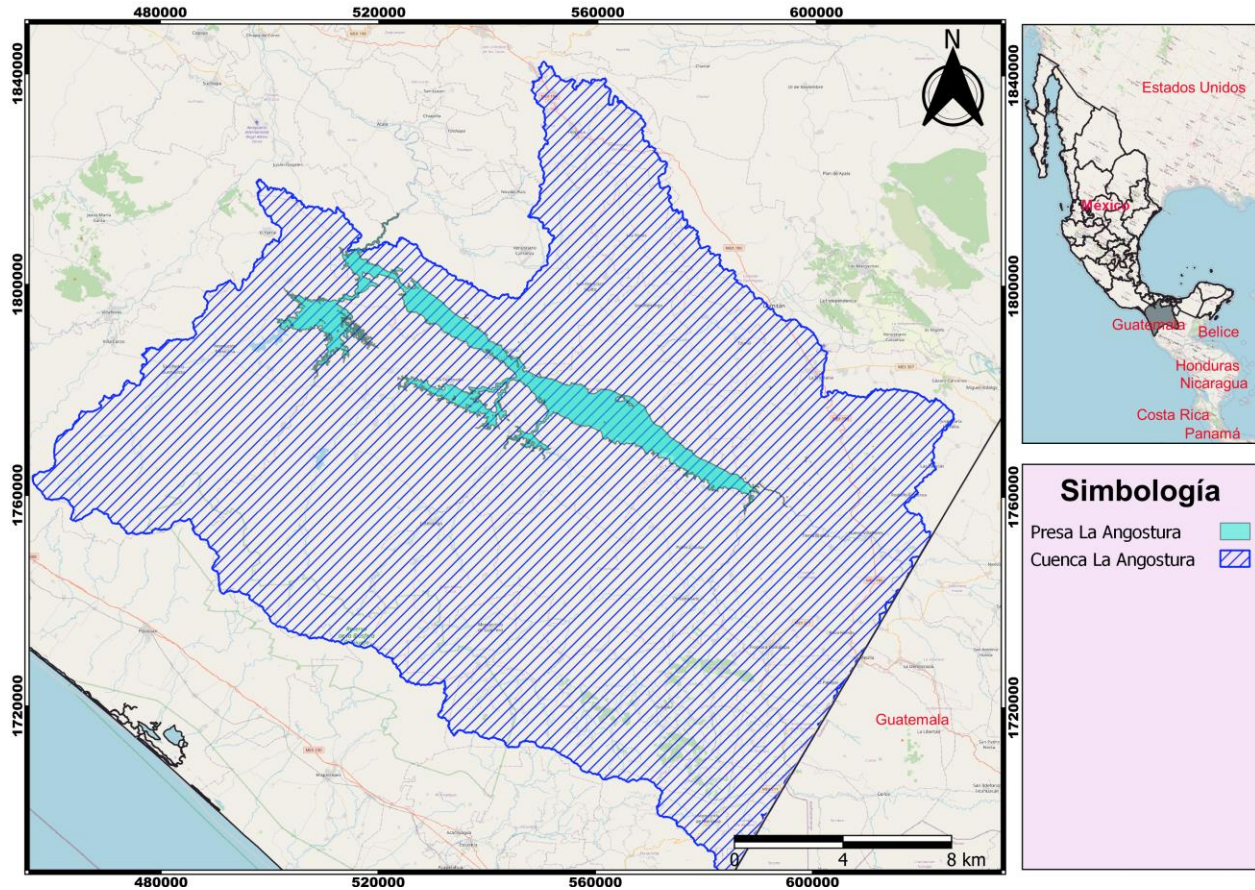


Figura 3.2. Ubicación de la zona de estudio (Elaboración propia).

3.1.2. Información disponible

CFE cuenta con más de 108 estaciones hidroclimatológicas que permiten medir de forma automática los niveles de agua en las presas, ríos y precipitaciones. La operación de la red de estaciones hidroclimatológicas de CFE tiene como finalidad el mejoramiento de la seguridad, disponibilidad y confiabilidad de todas las estructuras que conforman las centrales hidroeléctricas a nivel nacional y ayudar en proteger la población que se encuentra en las márgenes de inundación aguas debajo de las presas.

De otro lado, el Servicio Meteorológico Nacional de México (SMN) cuenta con 5500 estaciones climatológicas para registros diarios de lluvia, temperatura mínima y máxima en 24 horas. Permiten la elaboración de gráficas (ciclo anual y series temporales de temperaturas, precipitación, evaporación y unidades de calor).

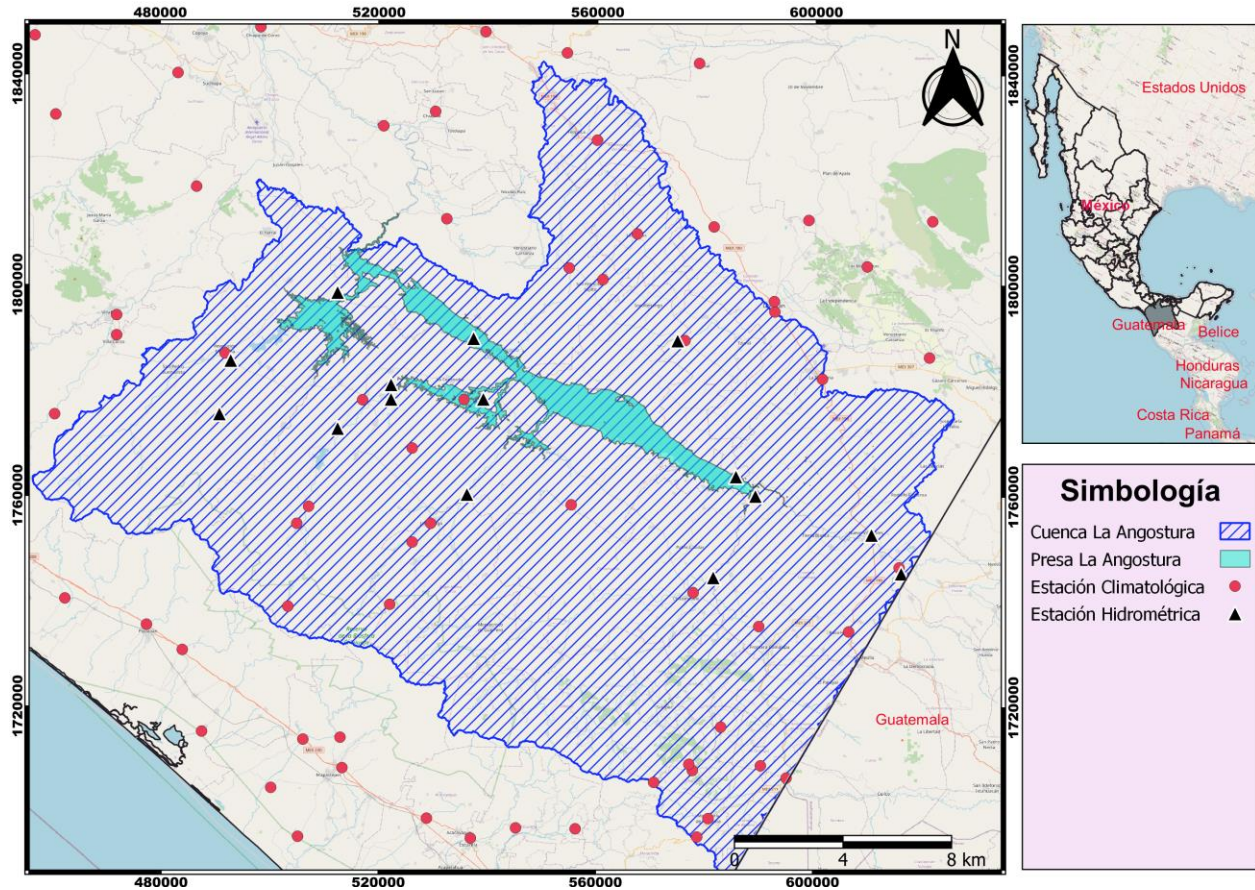


Figura 3.3. Información disponible en la zona de estudio.

La **Figura 3.3** muestra todas las estaciones hidrométricas de CFE que se encuentran dentro de la cuenca de la zona de estudio, varias estaciones climatológicas de (SMN), estaciones climatológicas que tienen un área de influencia sobre la cuenca, estaciones hidrométricas del Banco Nacional de Datos de Aguas Superficiales (BANDAS) y otras estaciones hidrométricas de la región de Angostura. La **Figura 3.4** muestra las tres estaciones hidrométricas seleccionadas y sus respectivas cuencas que se ubican dentro de la región de Angostura para el análisis de los modelos.

Para realizar análisis de los modelos de redes neuronales artificiales, autorregresivos y autorregresivos e integrados de medios móviles se eligieron las series temporales de las estaciones hidrométricas operada por la Comisión Federal de Electricidad, debido a la disponibilidad de los datos y las estaciones tuvieron un mínimo de 31 años de registros de periodo común (**Tabla 3.2**).

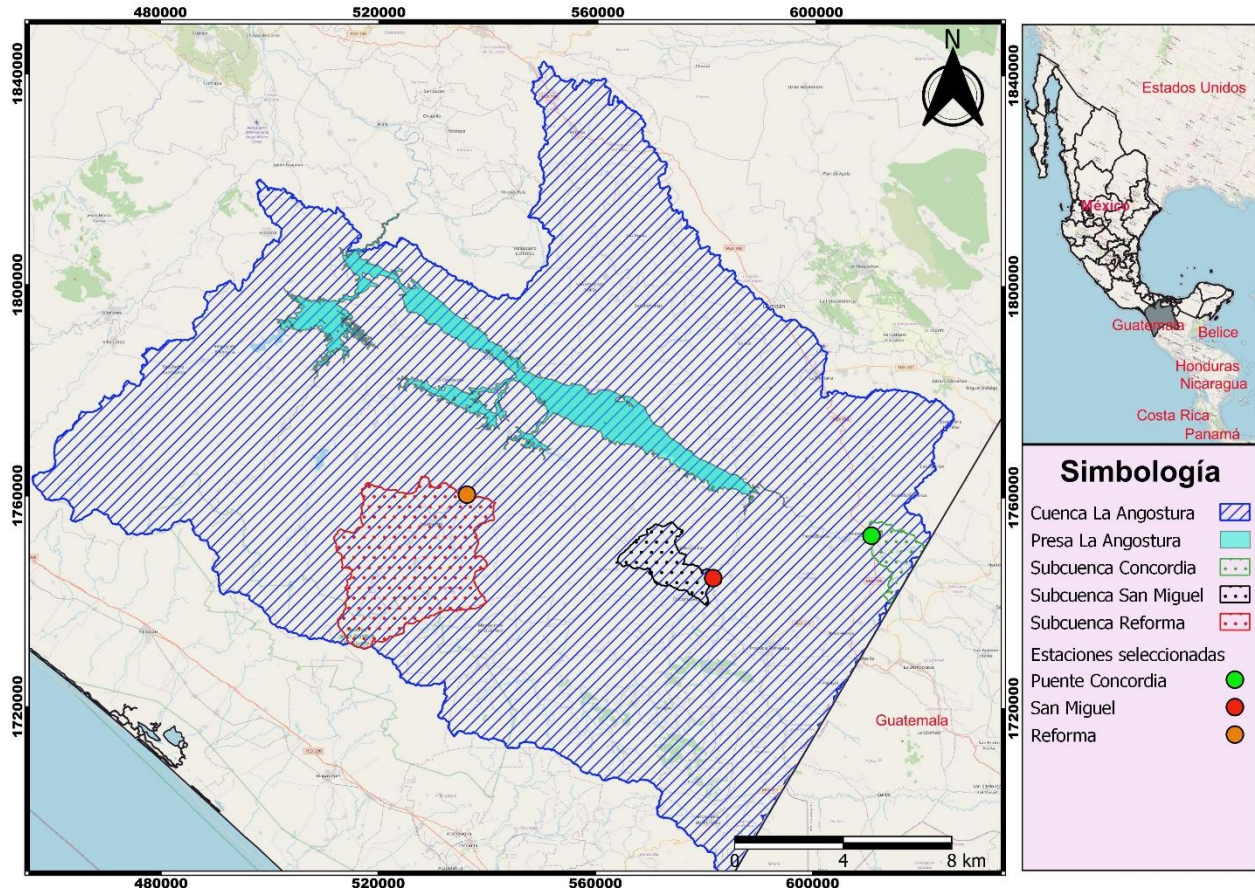


Figura 3.4. Estaciones hidrométricas seleccionadas y sus respectivas subcuencas.

La obtención de la información en las estaciones hidrométricas permite obtener datos confiables y de calidad para la aplicación de la metodología con finalidad de planeación y buen manejo de las aportaciones de escurrimientos en el embalse de la presa. Para un buen manejo de política de operación para evitar posibles derrames por vertedor y maximizar el aprovechamiento del agua para la generación de energía hidroeléctrica.

La **Tabla 3.2** muestra la información hidrométrica empleada en este trabajo de tesis para cada estación hidrométrica que se encuentra en la zona de estudio, son tres estaciones hidrométricas que tienen un periodo común de 31 años de registros de datos y la variable hidroclimatológica utilizada es el caudal medio diario de las estaciones que se ubican aguas arriba de la presa.

Tabla 3.2. Información hidrométrica utilizada.

| Estación | Longitud | Latitud | Parámetro | Periodo común | Años de Registro |
|------------------|-------------|-------------|---------------------|---------------|------------------|
| Puente Concordia | 91°58'04" O | 15°51'02" N | Caudal Medio Diario | 1977-2007 | 31 |
| San Miguel | 92°14'21" O | 15°46'40" N | Caudal Medio Diario | 1977-2007 | 31 |
| Reforma | 92°39'42" O | 15°55'15" N | Caudal Medio Diario | 1977-2007 | 31 |

3.1.3. Procesamiento de los datos

Características estadísticas

Se utilizaron las ecuaciones (2.115 a (2.122, (2.125 y (2.126 para determinar las características de las series temporales de los datos de caudal medio diario (m^3/s) de cada estación de la **Tabla 3.2** y se incorporó una nueva serie que es la suma de las tres estaciones hidrométricas seleccionadas de aguas arriba de la presa (**Figura 3.4**).

Tabla 3.3. Características estadísticas de las estaciones hidrométricas.

| Parámetros | Puente concordia | San Miguel | Reforma | Suma |
|-----------------------------------|------------------|------------|----------|-----------|
| Media (\bar{X}) | 72.99 | 112.31 | 24.92 | 210.22 |
| Varianza (S^2) | 8 659.26 | 37 070.12 | 1 014.84 | 92 340.18 |
| Desviación estándar (S) | 93.06 | 192.54 | 31.86 | 303.88 |
| Coficiente de Asimetría (g) | 22.39 | 24.04 | 3.58 | 21.88 |
| Curtosis (K) | 1 069.49 | 1 173.55 | 31.64 | 1 029.86 |
| Coficiente de variación (C_V) | 1.27 | 1.71 | 1.28 | 1.45 |
| Máximo (X_n) | 5 282.41 | 11 143.00 | 606.72 | 17 032.13 |
| Mínimo (X_1) | 2.00 | 1.00 | 0.37 | 21.90 |
| Mediana (M_o) | 49.10 | 54.40 | 12.91 | 123.89 |
| Número de días (N) | 11 322 | 11 322 | 11 322 | 11 322 |
| Rango (R) | 5 280.41 | 11 142.00 | 606.35 | 17 010.23 |

3.2. MODELOS ESTOCÁSTICOS

La metodología utilizada para la elección de los tres mejores modelos estocásticos autorregresivos (1 AR, 1 ARMA y 1 ARIMA) para predicción de gastos medio diario en la presa La Angostura, se basó en seleccionar el menor valor de criterio de información

de Akaike (AIC) y criterio de información Bayesiano de Schwartz (BIC) con las ecuaciones (2.31) y (2.32). Posteriormente se utilizaron estos modelos para predecir gastos del subconjunto de datos de entrenamiento (10592 datos, 93.55%), para validar cada modelo se evaluaron los valores pronosticados mediante índices de evaluación de pronóstico tales como: raíz del error medio cuadrático *RMSE* (en inglés Root Mean Square Error), error medio absoluto *MAE* (en inglés Mean Absolute Error), error medio absoluto porcentual *MAPE* (en inglés Mean Absolute Percentage Error) y el coeficiente de correlación *R* utilizando las ecuaciones (2.11) a (2.13) y (2.16).

La **Figura 3.5** muestra los gastos máximos anuales de cada estación hidrométrica de la zona de estudio a fin de observar el comportamiento de los datos en aguas arriba de la presa La Angostura.

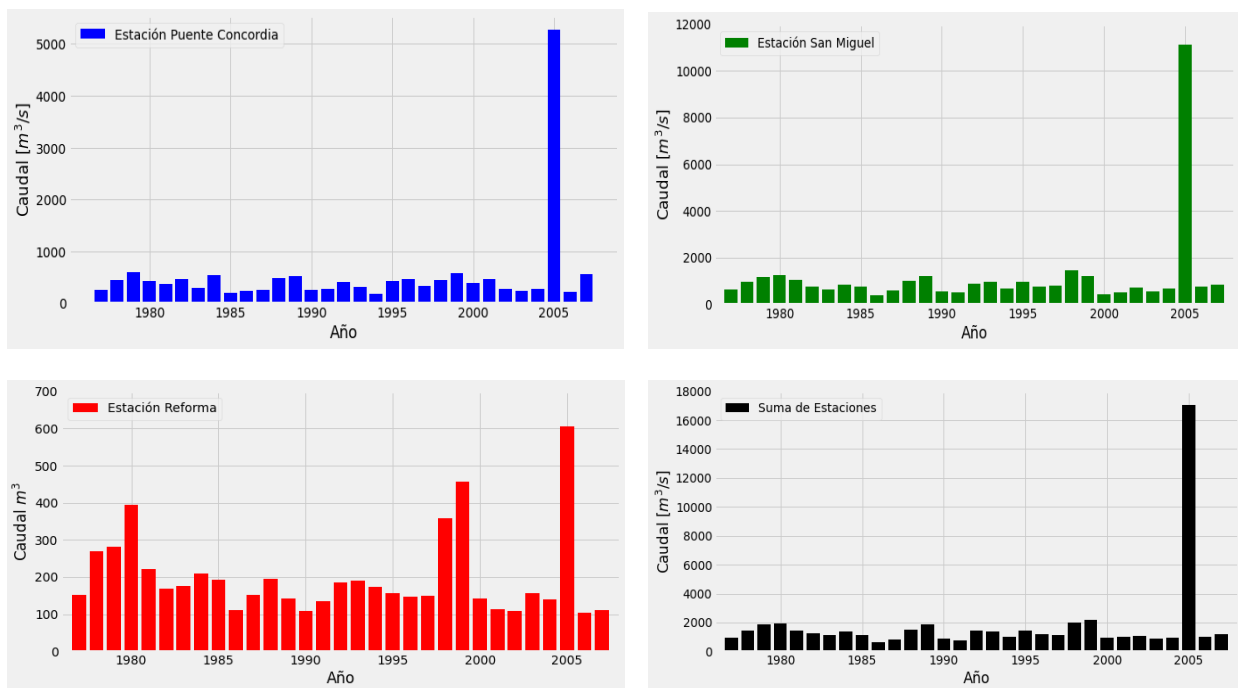


Figura 3.5. Máximos anuales de estaciones hidrométricas.

3.2.1. Prueba Dickey-Fuller Aumentada (ADF)

Se aplicó la prueba Dickey-Fuller Aumentada a la serie temporal de la suma de las tres estaciones de los datos originales para determinar el valor del estadístico *p*, la **Figura 3.8** muestra las gráficas de los datos originales, las de autocorrelación y autocorrelación

parcial. Pero la gráfica de autocorrelación muestra que todos los coeficientes están fuera de los límites de confianza, aunque el valor estadístico de la prueba $p \cong 0$, lo que es menor al 5% del nivel de significancia, pero se observó que la serie se está decreciendo lentamente, es decir, aun no es totalmente estacionaria.

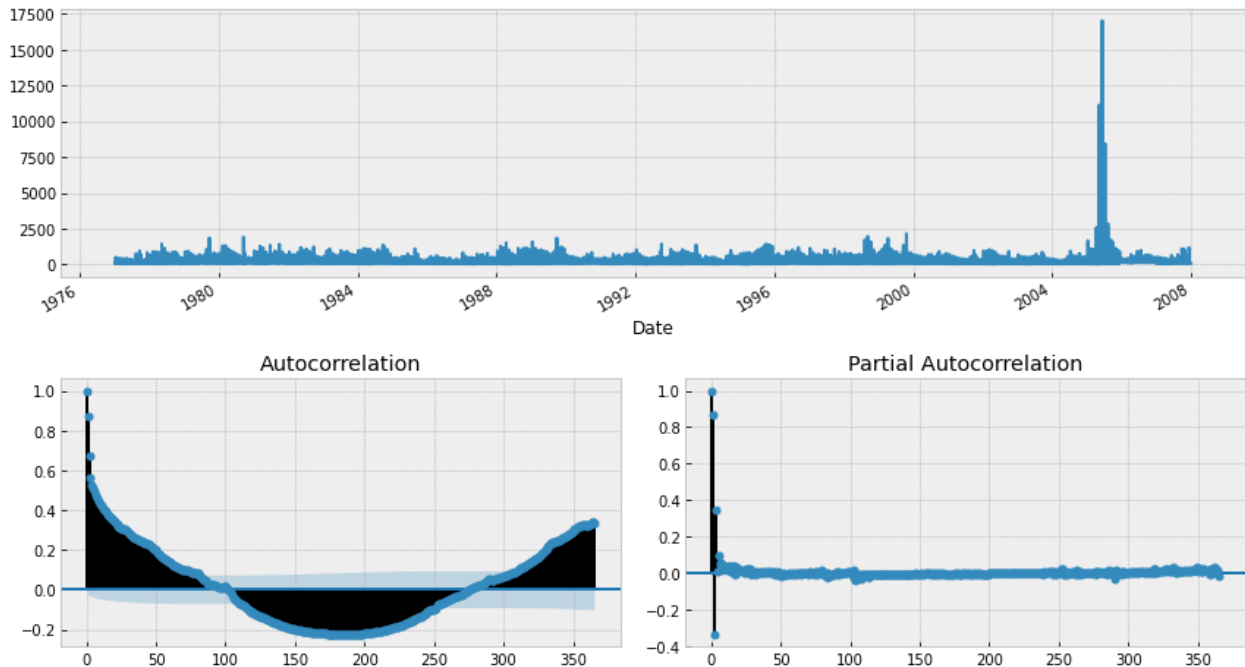


Figura 3.6. Prueba Dickey-Fuller Aumentada con datos originales y con retraso 365.

Según la prueba se itera los datos calculando la diferencia, hasta que se rechace la hipótesis nula y que los datos sean estacionarios. Entonces se diferencié la serie por una primera vez $d = 1$ para que sea estacionaria con media igual a cero y varianzas constantes, lo que se muestra en la **Figura 3.7** solamente con una primera diferenciación fue suficiente para que la serie sea estacionaria. El valor d es igual al número de veces que se ha diferenciado la serie temporal para que sea estacionaria mediante la prueba de Dickey-Fuller aumentada (ADF).

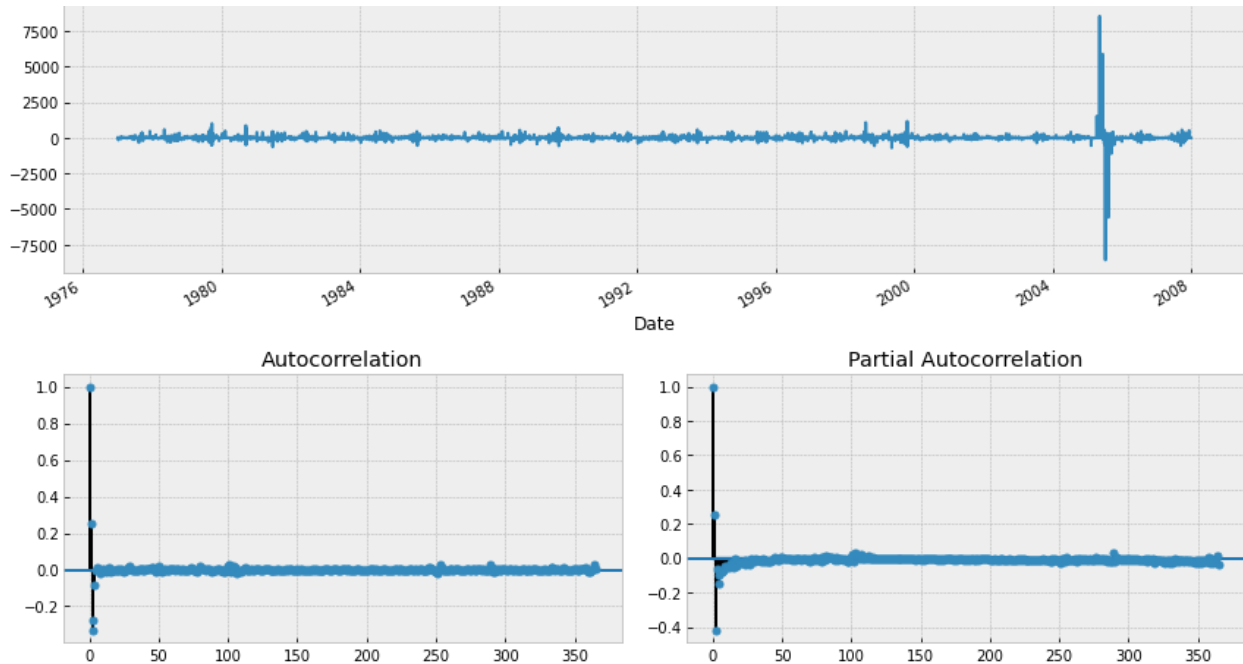


Figura 3.7. Prueba Dickey-Fuller Aumentada con datos $d=1$ y con retraso 365.

3.2.2. Separación de los datos

Se consideró un periodo común entre los datos de las estaciones hidrométricas están comprendido entre primero de enero de 1977 a 31 diciembre de 2007. Se dividió al conjunto de datos en dos subconjuntos: el primero se denomina datos de entrenamiento que comprende los registros desde el año 1977 hasta al año 2005 (10592 datos, 93.55%) y el segundo subconjunto fue nombrado datos de validación que va desde el año 2006 hasta al año 2007 (730 datos, 6.45%). Se utilizó este último durante el proceso de validación de resultados para probar el desempeño de los modelos y verificar problema de sobreajuste.

La **Figura 3.8** muestra la separación de los datos de cada estación hidrométrica en dos subconjuntos de datos de entrenamiento y validación.

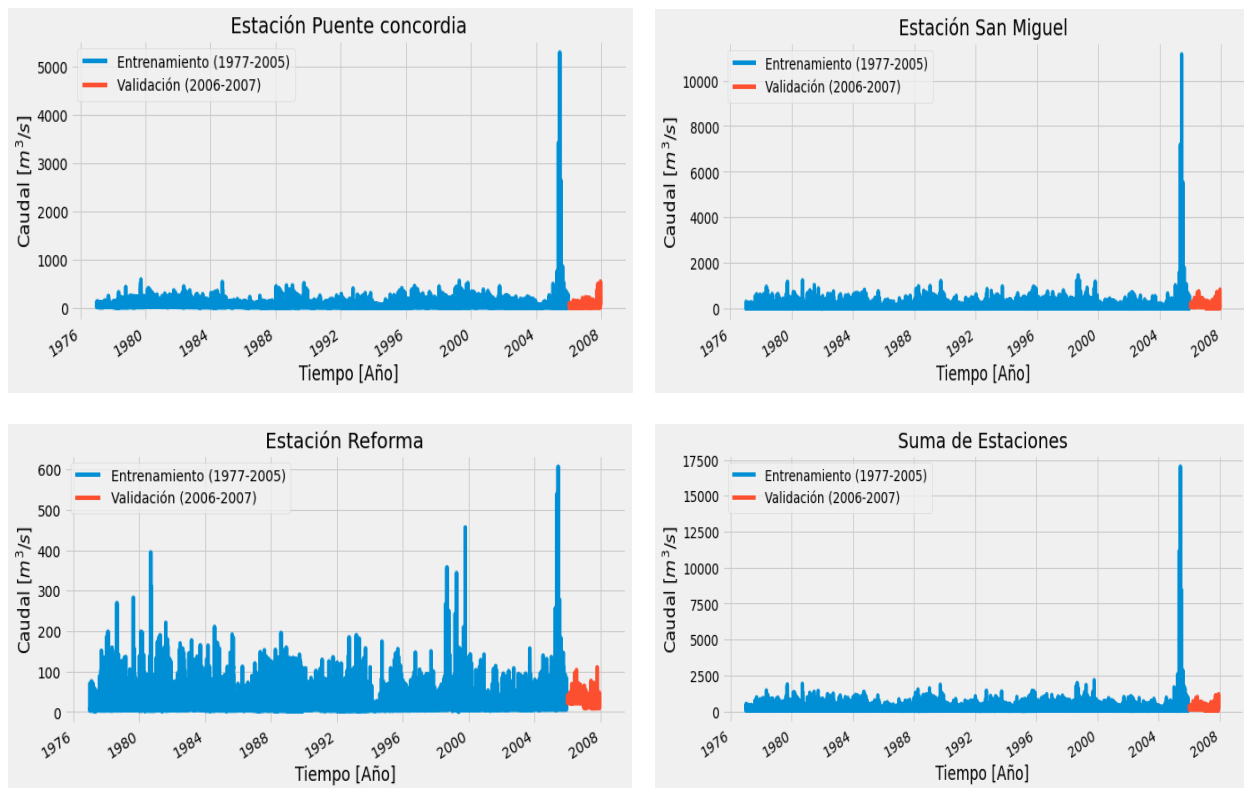


Figura 3.8. Separación de los datos de las estaciones hidrométricas.

3.2.3. Selección del mejor modelo estocástico

Se realizaron el análisis de las series de tiempos de las estaciones (solamente la **Suma**) de la **Tabla 3.3** aplicando la metodología de los modelos de AR(p), ARMA(p,q) y ARIMA(p,d,q) visto en el **CAPÍTULO 2:**, para obtener el mejor modelo estocástico autorregresivo según los valores de criterios de información de Akaike y Bayesiano.

Mejor modelo AR(p)

Para encontrar el número de términos p del modelo AR, se cuenta el número de rezagos que cruzan el límite de significancia en la gráfica de autocorrelación parcial (PACF). Se consideró un límite de confianza de 95% de los datos, se observa que los números de términos de p son $p = 1,2,4$ pero para evitar que la serie pudiera tener problemas de sobre o sub diferenciados se tomaron valores $p = 1,2,3,4$. La **Figura 3.9**

muestra la gráfica de autocorrelación parcial de los datos de entrenamiento con un límite de confianza de 95%.

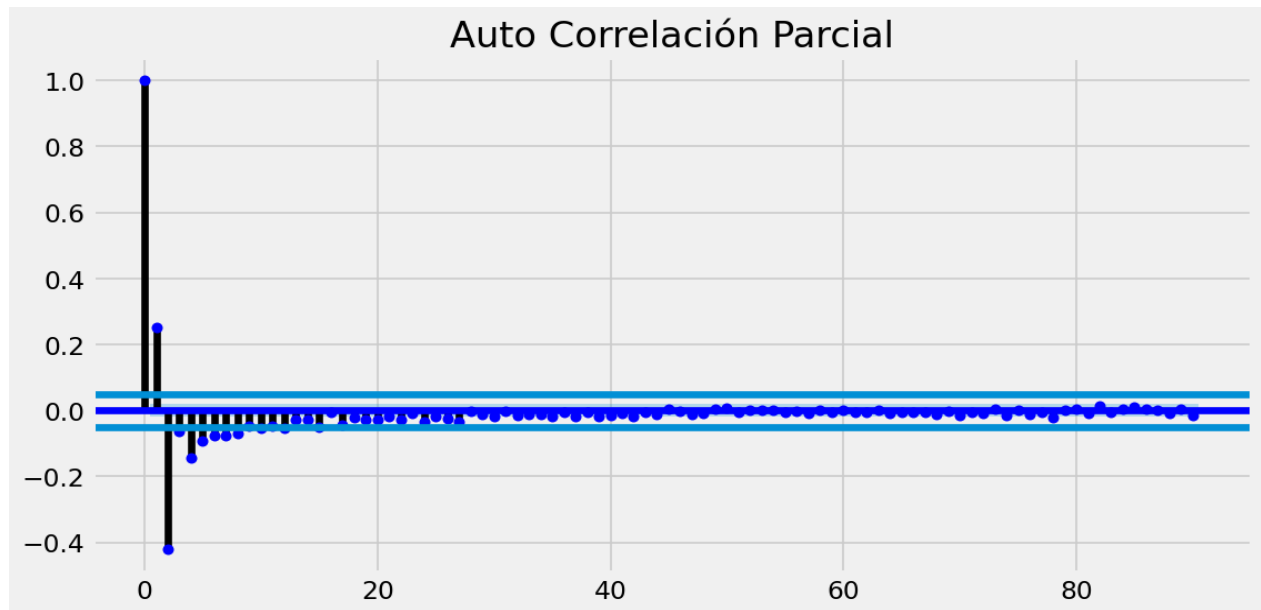


Figura 3.9. Gráfica PACF con retraso 90.

Luego de estos modelos AR(p) se eligieron los tres mejores según los valores de AIC y BIC, esos criterios se calcularon con el subconjunto de datos de entrenamiento de la serie temporal y los valores de los parámetros que se muestran en la **Tabla 3.4**.

La **Tabla 3.4** muestra los valores de los parámetros y criterios de los tres mejores modelos autorregresivos AR(p) obtenidos según los valores de los criterios AIC y BIC.

Tabla 3.4. Parámetros de modelos autorregresivos.

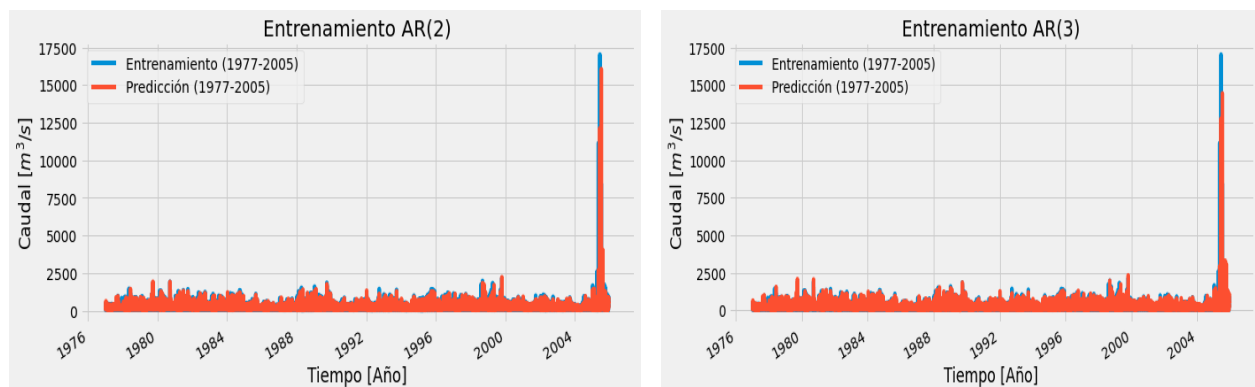
| PARÁMETROS | Modelos Autorregresivos AR(p) | | |
|---|-------------------------------|-------------|-------------|
| | AR(2) | AR(3) | AR(4) |
| Parámetro ϕ_1 | 1.1652 | 1.2854 | 1.2839 |
| Parámetro ϕ_2 | -0.3427 | -0.7514 | -0.7482 |
| Parámetro ϕ_3 | | 0.3507 | 0.3452 |
| Parámetro ϕ_4 | | | 0.0043 |
| Constante | 208.429 | 208.429 | 208.429 |
| Criterio de información Akaike <i>AIC</i> | 135 479.667 | 134 091.377 | 134 093.185 |
| Criterio Bayesiano de Schwartz <i>BIC</i> | 135 508.738 | 134 127.716 | 134 136.792 |

Luego con estos modelos auto regresivos de la **Tabla 3.4** se utilizaron para predecir los valores del entrenamiento, se consideró un paso de tiempo de un día, es decir con el valor de caudal medio del día i se calculó el valor de la predicción del día siguiente es decir $i + 1$ este mismo procedimiento se repite para los demás modelos estocásticos. Posteriormente, se determinaron los valores de índices de pronósticos para evaluar y comprar el desempeño de los modelos autorregresivos para la predicción de caudal en la fase del entrenamiento.

Tabla 3.5. Índices de comparación de modelos $AR(p)$.

| MODELO | R | RMSE | MAE | MAPE [%] |
|---------|-------|---------|--------|----------|
| $AR(2)$ | 0.884 | 144.893 | 43.973 | 33.249 |
| $AR(3)$ | 0.899 | 135.687 | 40.804 | 25.055 |
| $AR(4)$ | 0.899 | 135.686 | 40.741 | 24.979 |

La **Tabla 3.5** muestra los valores de índices de pronósticos, aunque en la **Tabla 3.4** el modelo $AR(3)$ tiene menor valor de AIC y BIC pero para predicción de los datos de entrenamiento el modelo $AR(4)$ dio mejores resultados. Menor $RMSE$ (135.686), menor MAE (40.741) y menor $MAPE$ (24.979) entonces el mejor modelo autorregresivo es $AR(4)$. La **Figura 3.10** muestra el conjunto de datos de entrenamiento con su respectiva predicción de cada uno de los tres mejores modelos autorregresivos $AR(p)$.



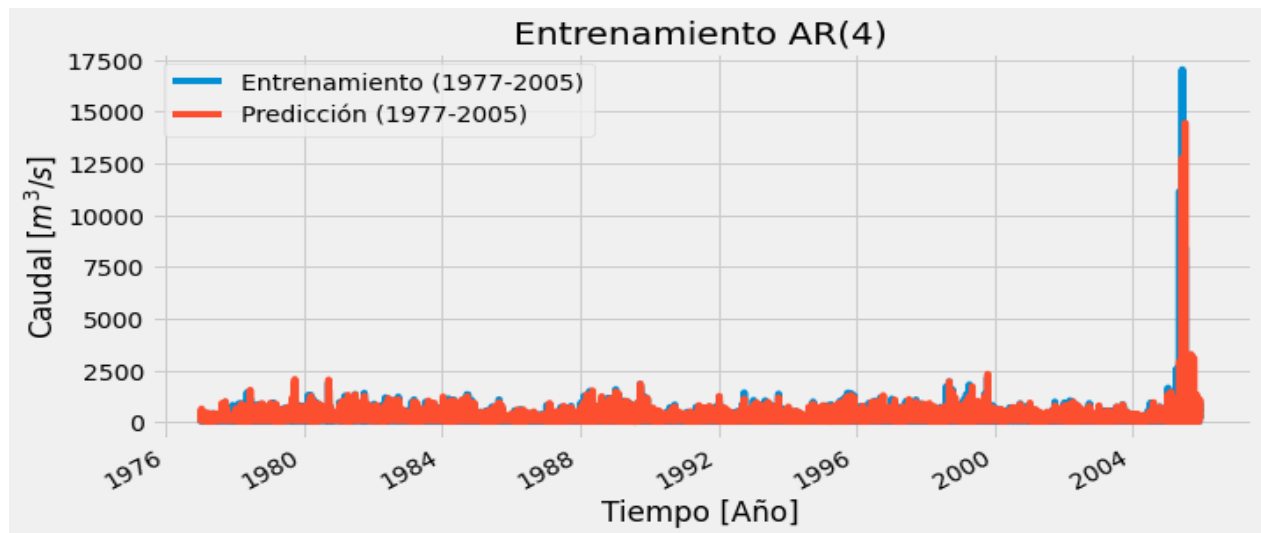


Figura 3.10. Pronósticos de la fase de entrenamiento de $AR(p)$.

Mejor modelo ARMA(p,q)

De la misma manera, para encontrar el número de términos q de la media móvil MA(q), se contó el número de rezagos que cruzaron el límite de significancia en la gráfica de autocorrelación (ACF).

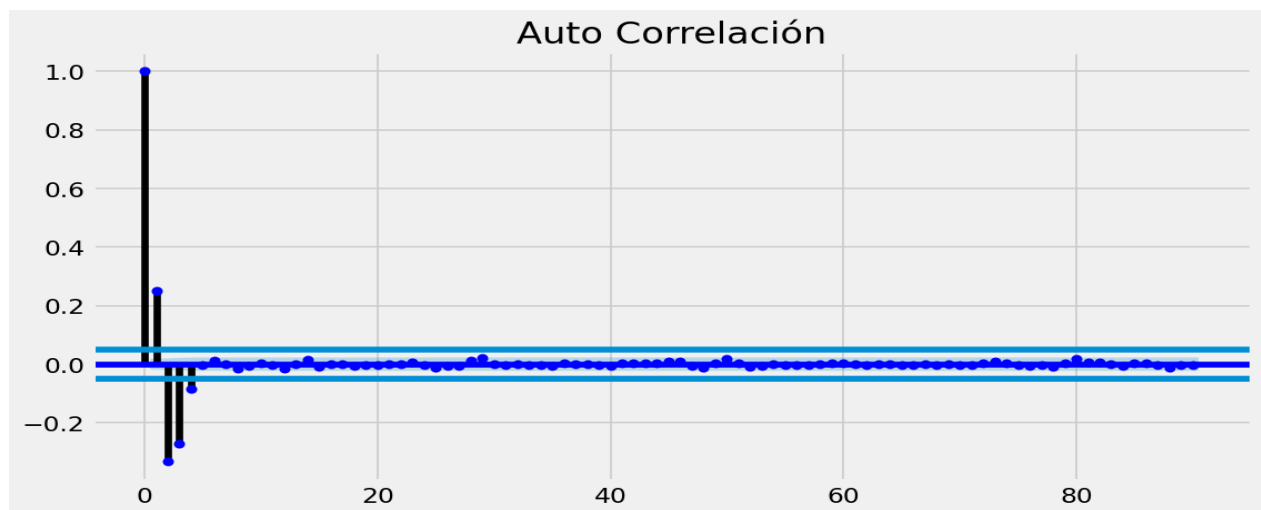


Figura 3.11. Gráfica ACF con retraso 90.

La **Figura 3.11** muestra la gráfica ACF que indica el número de términos de media móvil que se requiere para eliminar cualquier autocorrelación en la serie estacionalizada. Igualmente, se consideraron un límite de confianza de 95% de los datos **Figura 3.9**, se

observa que los valores $q = 1,2,3,4$ se encuentran fuera de los límites de confianza entonces son los términos de la media móvil.

La **Tabla 3.6** muestra los valores de los parámetros autorregresivos y medios móviles de cada modelo $ARMA(p,q)$. Como se hizo anteriormente, aparte de los criterios de AIC y BIC se calcularon los valores de predicción de caudal medio de los datos de entrenamiento utilizando cada uno de los modelos $ARMA$ y se evaluaron sus desempeños mediante los valores de índices de pronósticos con mucho énfasis sobre el valor del error porcentual medio absoluto.

Tabla 3.6. Parámetros de modelos autorregresivos de medios móviles.

| PARÁMETROS | Modelos Autorregresivo de Media Móvil $ARMA(p,q)$ | | |
|--------------------------------------|---|-------------|-------------|
| | $ARMA(1,4)$ | $ARMA(2,3)$ | $ARMA(2,4)$ |
| Parámetro ϕ_1 | 0.9726 | 1.2977 | 1.1352 |
| Parámetro ϕ_2 | | -0.3135 | -0.1564 |
| Parámetro θ_1 | 0.2918 | -0.0329 | 0.1307 |
| Parámetro θ_2 | -0.4125 | -0.5035 | -0.4563 |
| Parámetro θ_3 | -0.3526 | -0.2087 | -0.2822 |
| Parámetro θ_4 | -0.1109 | | -0.0613 |
| Constante | 208.429 | 208.429 | 208.430 |
| Criterio de información Akaike AIC | 133 838.217 | 133 838.765 | 133 837.315 |
| Criterio Bayesiano de Schwartz BIC | 133 889.092 | 133 889.640 | 133 895.458 |

La **Tabla 3.7** muestra los valores de índices de pronósticos de los mejores modelos autorregresivos de media móvil.

Tabla 3.7. Índices de comparación de modelos $ARMA(p, q)$.

| MODELO | R | RMSE | MAE | MAPE [%] |
|-------------|-------|---------|--------|----------|
| $ARMA(1,4)$ | 0.901 | 134.048 | 39.539 | 19.342 |
| $ARMA(2,3)$ | 0.901 | 134.052 | 39.481 | 19.012 |
| $ARMA(2,4)$ | 0.901 | 134.030 | 39.510 | 19.169 |

La **Figura 3.12** muestra el subconjunto de datos de entrenamiento con su respectivo valor de predicción de cada uno de los tres mejores modelos autorregresivos de media

móvil ARMA(p,q). Con menor $MAE(39.481)$ y menor $MAPE(19.012)$ entonces el mejor modelo autorregresivo de media móvil es $ARMA(2,3)$.

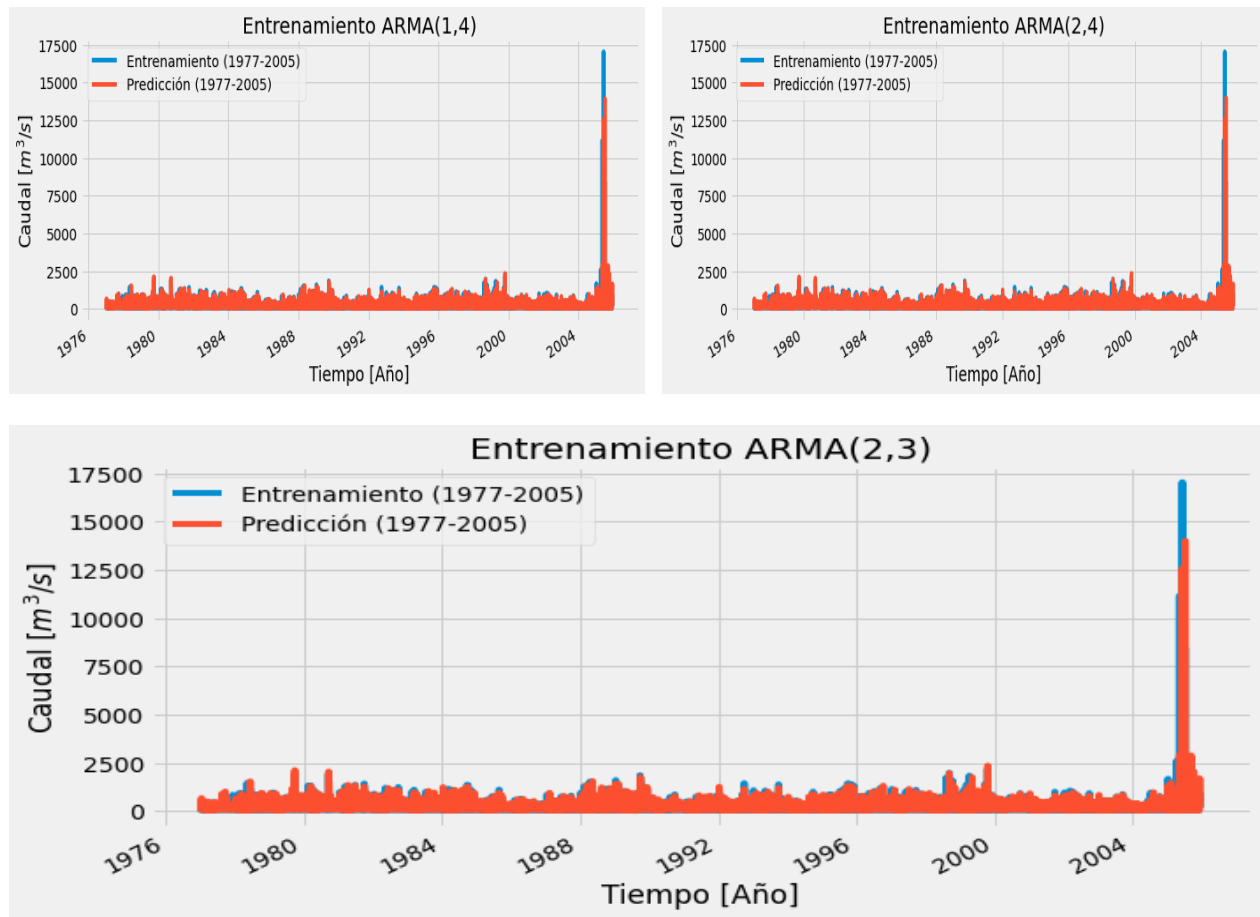


Figura 3.12. Pronósticos de la fase de entrenamiento de $ARMA(p, q)$.

Mejor modelo ARIMA(p,d,q)

Finalmente, para generar los modelos autorregresivos integrados de medias móviles se consideraron los valores de $p = 1,2,3,4$, $q = 1,2,3,4$ y $d = 1$ calculados anteriormente, no se consideraron la parte periódica (componente estacional) de los modelos ARIMA, es decir, $P = Q = D = 0$.

La **Tabla 3.8** muestra los valores de los parámetros de los tres mejores modelos autorregresivos integrados de medios móviles de cada modelo $ARIMA(p, d, q)$. De acuerdo con los valores de criterios AIC y BIC , el mejor modelo ARIMA es el

$ARIMA(3,1,3)$ tiene menor valor de criterios que los otros dos modelos. Y se calcularon los valores de índices de pronósticos que muestra en la **Tabla 3.9**.

Tabla 3.8. Parámetros de modelos auto regresivos integrados de medios móviles.

| PARÁMETROS | Modelos Autorregresivo Integrado de Media Móvil $ARMA(p,d,q)$ | | |
|--------------------------------------|---|----------------|----------------|
| | $ARIMA(3,1,2)$ | $ARIMA(4,1,2)$ | $ARIMA(3,1,3)$ |
| Parámetro ϕ_1 | 0.9310 | 0.7359 | 1.3396 |
| Parámetro ϕ_2 | -0.4262 | -0.1847 | -0.6994 |
| Parámetro ϕ_3 | 0.1539 | 0.0057 | 0.2072 |
| Parámetro ϕ_4 | | 0.0653 | |
| Parámetro θ_1 | -0.6569 | -0.4625 | -1.0667 |
| Parámetro θ_2 | -0.2589 | -0.4471 | -0.0998 |
| Parámetro θ_3 | | | 0.2014 |
| Criterio de información Akaike AIC | 133 899.933 | 133 897.765 | 133 889.993 |
| Criterio Bayesiano de Schwartz BIC | 133 943.539 | 133 948.640 | 133 940.867 |

Finalmente, estos tres mejores modelos autorregresivos integrados de medios móviles se usaron para predecir los valores del subconjunto de entrenamiento y se compararon estos valores entre sí, a fin de seleccionar el mejor modelo $ARIMA$.

Tabla 3.9. Índices de comparación de $ARIMA(p, d, q)$.

| MODELO | R | RMSE | MAE | MAPE [%] |
|----------------|-------|---------|--------|----------|
| $ARIMA(3,1,2)$ | 0.901 | 134.535 | 38.745 | 13.245 |
| $ARIMA(4,1,2)$ | 0.901 | 134.508 | 38.857 | 13.381 |
| $ARIMA(3,1,3)$ | 0.901 | 134.450 | 38.986 | 13.597 |

La **Figura 3.13** muestra el subconjunto de datos de entrenamiento y el de predicción de cada uno de los modelos autorregresivos integrados de media móvil $ARMA(p,d,q)$. Con menor $RMSE$ (134.450) entonces el mejor modelo $ARIMA$ es el $ARIMA(3,1,3)$.

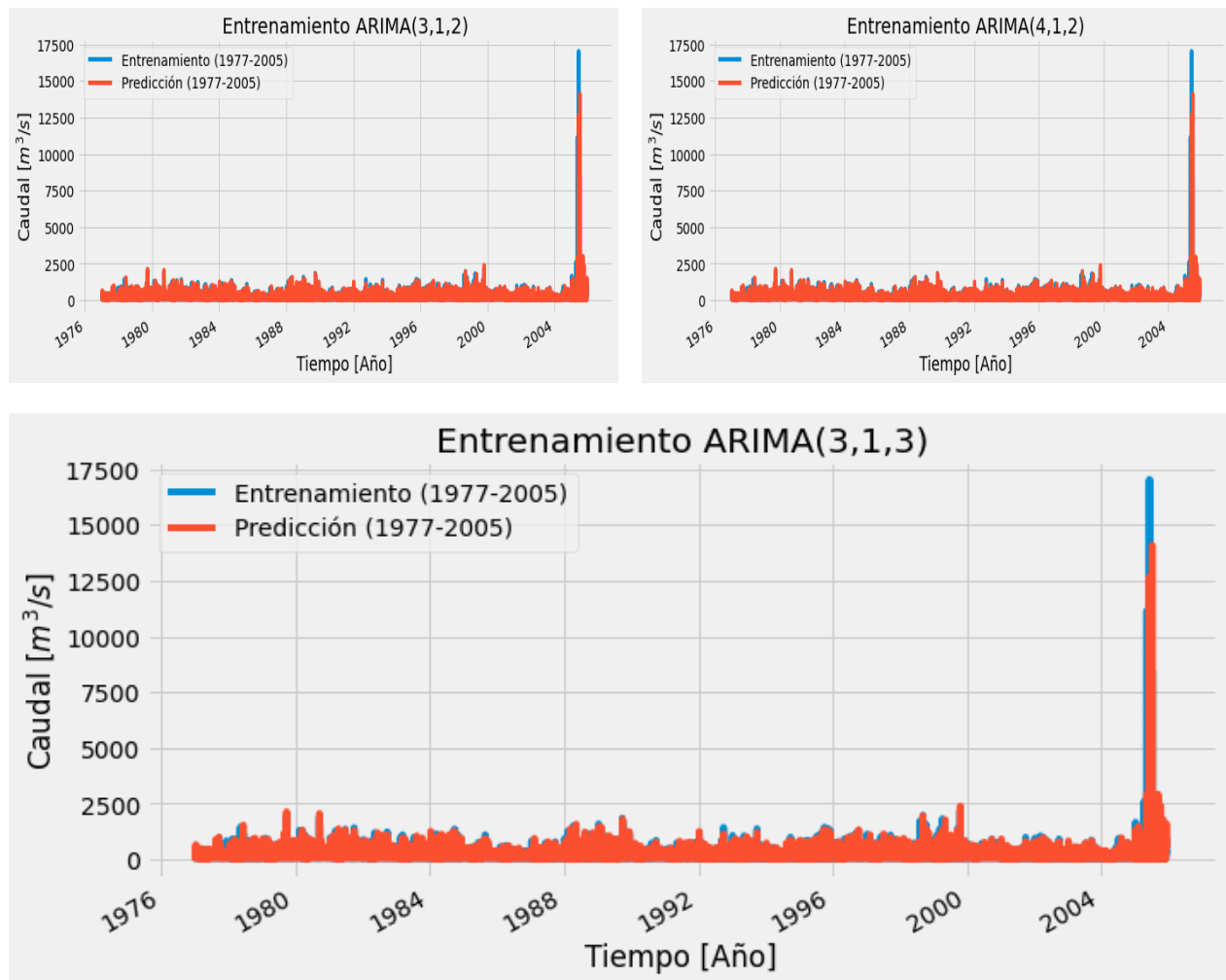


Figura 3.13. Pronósticos de la fase de entrenamiento de $ARIMA(p, d, q)$.

Finalmente, la **Tabla 3.10** muestra la selección de los 3 mejores modelos estocásticos autorregresivos, se eligió el mejor modelo de auto regresivo $AR(p)$, el mejor modelo auto regresivo de media móvil $ARMA(p, q)$ y el mejor modelo autorregresivo integrado de media móvil $ARIMA(p, d, q)$. Se los utilizarán para predicción del subconjunto de datos en la fase de validación, mientras en la fase de entrenamiento el modelo $ARIMA(3,1,3)$ tiene menor valor de $MAE(38.986)$ y menor $MAPE(13.597)$ y un coeficiente de correlación de $R(0.901)$ que indica una buena correlación entre los valores de caudales reales y pronosticados entonces $ARIMA(3,1,3)$ es el mejor modelo estocástico autorregresivo para predicción de datos en la fase de entrenamiento, pero se

consideraron los tres modelos de la **Tabla 3.10** para la predicción de los datos de caudales medios en la fase de validación.

Tabla 3.10. Índices de pronósticos del entrenamiento de modelos estocásticos.

| MODELO | R | RMSE | MAE | MAPE [%] |
|---------------------|-------|---------|--------|----------|
| <i>AR(4)</i> | 0.899 | 135.686 | 40.741 | 24.979 |
| <i>ARMA(2,3)</i> | 0.901 | 134.052 | 39.481 | 19.012 |
| <i>ARIMA(3,1,3)</i> | 0.901 | 134.450 | 38.986 | 13.597 |

3.3. MODELOS DE REDES NEURONALES ARTIFICIALES

Los pasos para la selección de los tres mejores modelos de redes neuronales artificiales para predicción de caudales medio diarios, se puede resumir así: se construye, se entrena y se prueba la red neuronal utilizando los datos del subconjunto de la fase de entrenamiento. Se utilizó una RNA de tipo recurrente de memoria a corto plazo de LSTM (en inglés Long Short Term Memory). Se hizo uso de librerías de código abierto en el ambiente del lenguaje de programación Python.

Algunas librerías utilizadas son:

Pandas es una biblioteca de código abierto con licencia BSD (Berkeley Software Distribution) que proporciona estructuras de datos y herramientas de análisis de datos fáciles de usar y de alto rendimiento para el lenguaje de programación Python.

NumPy es el paquete fundamental para la computación científica en Python. Es una biblioteca de Python que proporciona un objeto de matriz multidimensional, el paquete incluye una gran variedad, manipulación matemática, lógica, de formas, clasificación, selección, transformadas discretas de Fourier, álgebra lineal básica, operaciones estadísticas básicas, simulación aleatoria y mucho más (NumPy community, 2022).

Matplotlib es una biblioteca completa que sirve para crear visualizaciones estáticas, animadas e interactivas en Python (Matplotlib, 2022).

- Crear parcelas de calidad de publicación.
- Hacer figuras interactivas que puedan hacer zoom, desplazarse y actualizar.
- Personalizar el estilo visual y el diseño.
- Exportar a muchos formatos de archivo.

TensorFlow es una plataforma de código abierto integral para el aprendizaje automático, facilita a principiantes y expertos la creación de modelos de aprendizaje automático. Es una biblioteca de software gratuita y de código abierto para el aprendizaje automático y la inteligencia artificial. Se puede utilizar en una variedad de tareas, pero tiene un enfoque particular en el entrenamiento y la inferencia de redes neuronales profundas. Es una plataforma de extremo a extremo enfocada en el aprendizaje automático, explora soluciones para acelerar las tareas de aprendizaje automático en todos los pasos del flujo de trabajo (Torres, 2020).

Escalamiento de los datos

Se transformaron los valores de las series temporales a valores [0,1], mediante la transformación de MinMaxScaler.

La fórmula de normalización:

$$(3.1) \quad y_i = \frac{x_i + x_{min}}{x_{max} - x_{min}}$$

x_{max} y x_{min} : valores máximo y mínimo de la serie respectivamente

y_i : valor normalizado o escalado

x_i : cualquier valor de la serie

Se normalizaron los dos subconjuntos de datos el de entrenamiento y validación respectivamente con la ecuación (3.1).

Diseño de la red

Se programó el modelo de la red neuronal en el lenguaje Python se utilizaron las librerías anteriormente mencionadas que son de código abierto. Para la ejecución del

código de la red neuronal se utilizó Google Colaboratory, que es una herramienta basada en Cloud Computing que permite tener acceso al uso de GPU (Graphic Processing Unit) y TPU (Tensor Processing Unit). Se utilizó la versión gratuita que cumplió los requerimientos de los modelos de las redes neuronales artificiales.

Ajuste de Hiperparámetros

Se realizó un análisis de ajuste de hiperparámetros para determinar la combinación correcta de los valores de búsqueda tales el número de capas ocultas, número de neuronas, tasa de aprendizaje, tamaño de lote de datos (batch size), número de épocas o iteraciones necesarias y función de activación. Se utilizó la búsqueda de cuadrícula, se entrenó las redes neuronales con el subconjunto de datos de entrenamiento, se evaluó el desempeño de cada combinación y se seleccionaron las mejores combinaciones. Se usaron tres iteraciones de validación cruzada (*cv*) que itera cada combinación tres veces.

La **Tabla 3.11** muestra un rango de los valores de búsqueda utilizado para el análisis de ajuste de hiperparámetros mediante el método de búsqueda de cuadrícula. Anteriormente se hizo uso del conocimiento que se tiene sobre el problema para realizar un análisis de prueba y error que permitió fijar un rango de valores para cada parámetro de búsqueda.

Se consideró un máximo de 3 capas ocultas ya que con más capas ocultas se requiere mucho consumo de memoria de la computadora para ejecución de la red, una capa oculta puede llevar a tener máximo 64 neuronas, también se consideró una tasa de aprendizaje máxima de 10% y tamaño de bloque o lote de 30, 90 y 180 lo que significa un lote de 30 datos se está dividiendo el año en sus doce meses, 90 datos en sus cuatro estaciones y 180 datos se tiene en cuenta los datos de las dos temporadas (lluviosa y seca) del año para realizar el entrenamiento.

Tabla 3.11. Valores de búsqueda del ajuste de hiperparámetros.

| PARÁMETROS DE BÚSQUEDA | VALORES | | |
|------------------------|---------|------|---------|
| Capas ocultas | 1 | 2 | 3 |
| Neuronas | 16 | 32 | 64 |
| Tasa de aprendizaje | 0.001 | 0.01 | 0.1 |
| Tamaño de bloque | 30 | 90 | 180 |
| Épocas | 2000 | 2000 | 2000 |
| Función activación | relu | tanh | softmax |

Se consideró cada combinación depende de la función de activación y cada combinación tiene 3 posibilidades entonces, el número total de combinación se calculó de la siguiente manera:

$$C = p^n C_v$$

C : número de combinación posible dependiendo de la función de activación

p : número de posibilidad de cada parámetro de búsqueda

n : número total de parámetros de búsqueda

C_v : número de cruce de validación

Entonces,

$$C = 3^5 * 3 = 3^6 = 729 \text{ ajustes}$$

Para cada función de activación se tuvo un total de 729 ajustes de hiperparámetros o combinaciones posibles, entonces fueron 3 funciones de activación por lo tanto se obtuvo un total de 2187 ajustes de hiperparámetros posibles de los cuales se seleccionaron las 3 mejores combinaciones para el pronóstico de escurrimientos en la fase de entrenamiento y validación de los datos. Posteriormente se procedió al entrenamiento de la red neuronal.

La **Tabla 3.12** muestra los valores de los parámetros de búsqueda de las tres mejores combinaciones o modelos LSTM encontrados utilizando el subconjunto de datos de la fase de entrenamiento con el método de búsqueda de cuadrícula para el ajuste de hiperparámetros. Cada modelo tiene solamente una capa oculta y diferentes valores para cada parámetro de búsqueda.

Tabla 3.12. Ajuste de hiperparámetros de las mejores combinaciones.

| PARÁMETROS DE BÚSQUEDA | LSTM_1 | LSTM_2 | LSTM_3 |
|------------------------|--------|--------|--------|
| Capas ocultas | 1 | 1 | 1 |
| Neuronas | 32 | 32 | 64 |
| Tasa de aprendizaje | 0.001 | 0.001 | 0.001 |
| Tamaño de bloque | 30 | 30 | 180 |
| Épocas | 2000 | 2000 | 2000 |
| Función activación | tanh | relu | tanh |

Entrenamiento

Con los datos de la **Tabla 3.12** se entrenó cada uno de los modelos, se realizó mediante la técnica de descenso del gradiente con sus respectivos tamaños de bloque o lote de datos, se utilizó la función de coste del error medio cuadrático (MSE) de la ecuación (2.112) y el algoritmo Adam para minimizar la función de coste y ajustar la tasa de aprendizaje durante el entrenamiento.

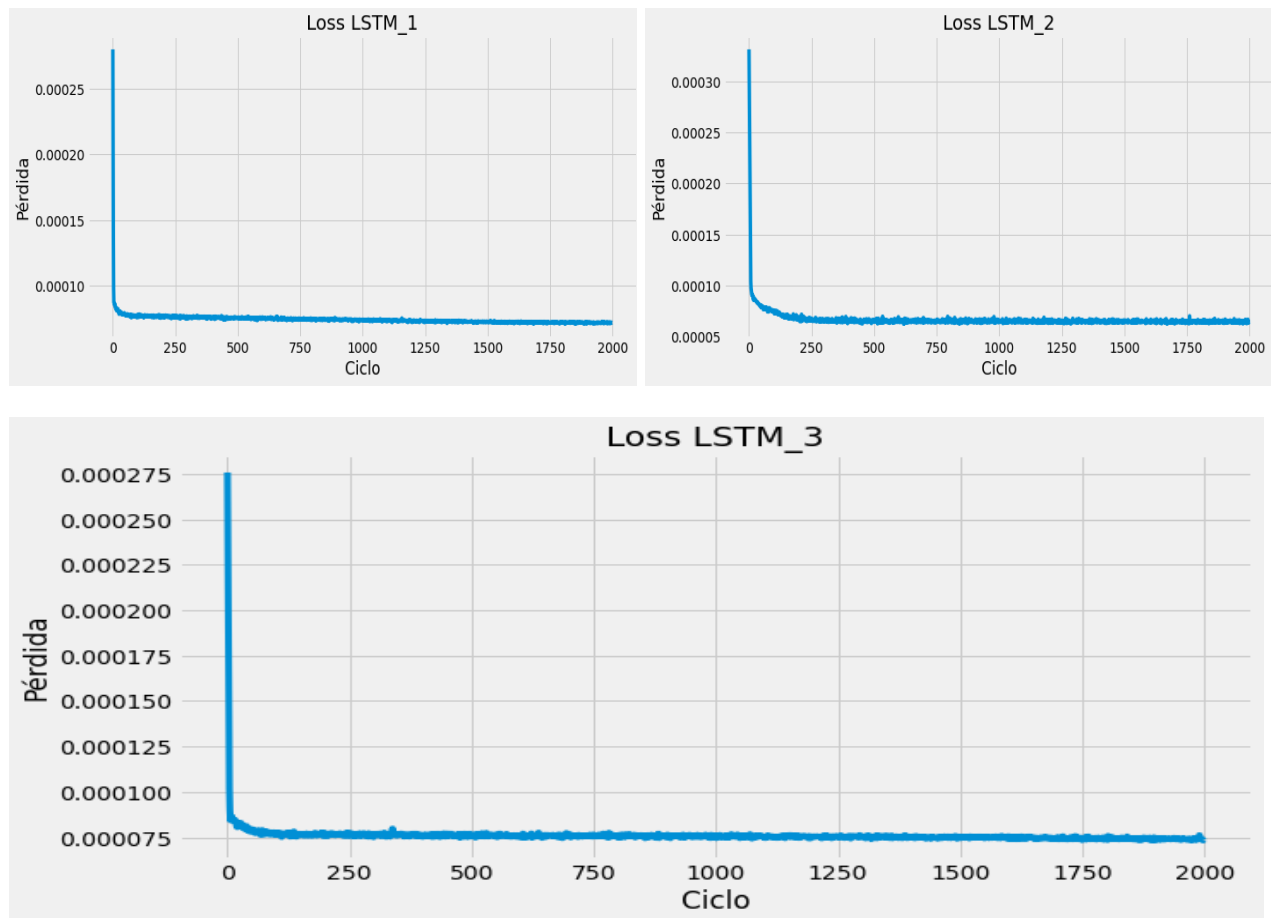


Figura 3.14. Función de coste de cada modelo de red neuronal.

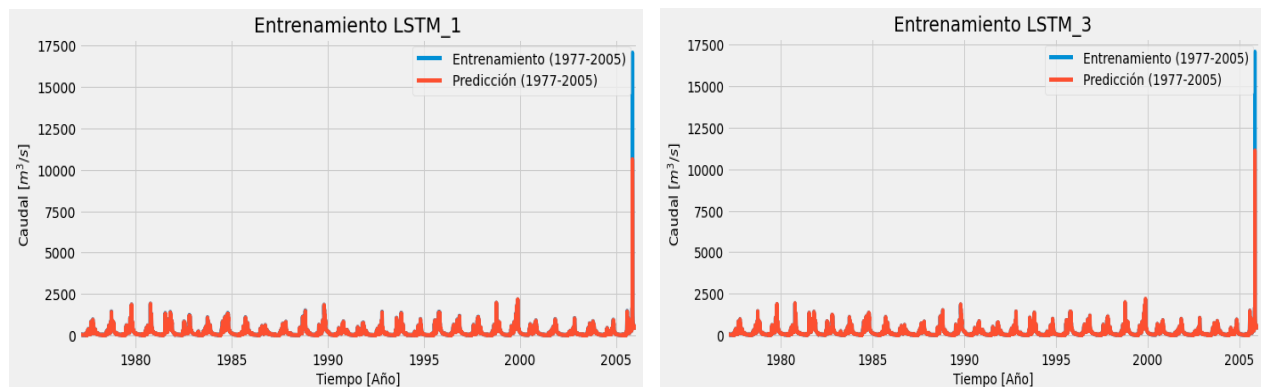
La **Figura 3.14** muestra la función de coste de cada modelo de red neuronal calculado a partir del número de época mediante el error medio cuadrático durante la fase de entrenamiento de la red neuronal.

Tabla 3.13. Índices de pronósticos de escenarios de la red neuronal.

| MODELO | R | RMSE | MAE | MAPE [%] |
|---------------|-------|---------|--------|----------|
| <i>LSTM_1</i> | 0.889 | 137.614 | 30.465 | 9.675 |
| <i>LSTM_2</i> | 0.900 | 134.869 | 29.833 | 8.791 |
| <i>LSTM_3</i> | 0.898 | 136.859 | 31.219 | 10.007 |

Como se mencionó anteriormente, se utilizaron los modelos de redes neuronales artificiales recurrentes para pronosticar los valores del subconjunto de datos de caudales del entrenamiento en la presa. La **Tabla 3.13** muestra los valores de índices de pronósticos de los tres mejores modelos de redes neuronales artificiales recurrentes obtenidos.

La **Figura 3.15** muestra el conjunto de datos de entrenamiento y su respectiva predicción de cada uno de los tres modelos de redes neuronales artificiales recurrentes. Con menor *RMSE*(134.869), menor *MAE*(29.833), menor *MAPE*(8.791) y mayor coeficiente de correlación *R*(0.900) entonces el mejor modelo de red neuronal artificial recurrente es *LSTM_2*.



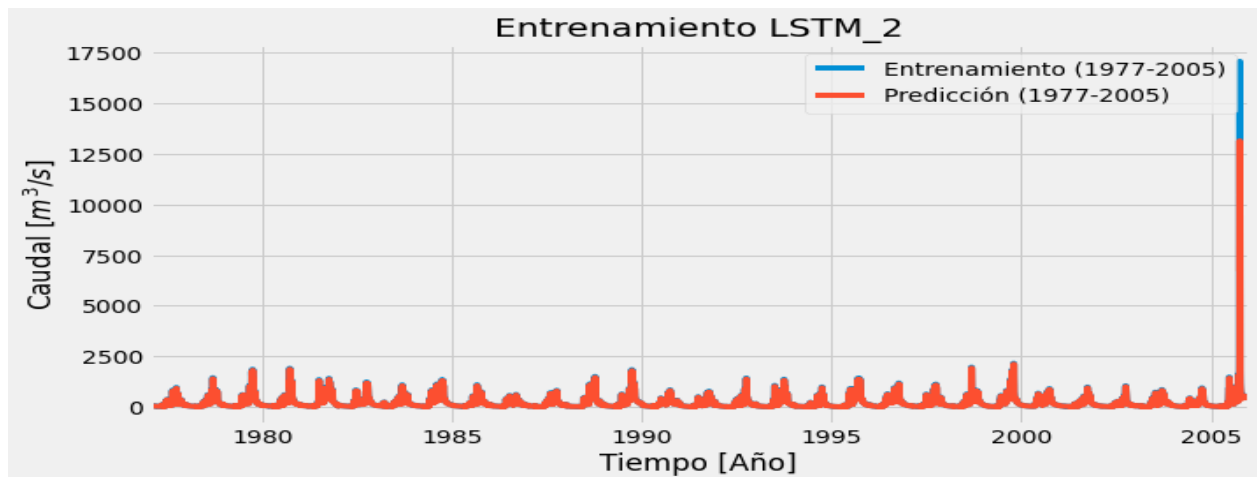


Figura 3.15. Pronósticos de la fase de entrenamiento de modelos de red neuronal *LSTM*.

Finalmente, el mejor modelo de redes neuronales artificiales recurrentes en la fase de entrenamiento de los datos es *LSTM_2* contiene una capa oculta intermedia de 32 neuronas, una tasa de aprendizaje de 0.1%, tamaño de lote de 30 datos, 2000 épocas y la función de activación *relu*.

CAPÍTULO 4:

4. DISCUSIONES DE RESULTADOS

En este capítulo se analizaron los resultados de pronósticos obtenidos del subconjunto de datos de validación (test), primero se pronosticaron los datos de validación con los modelos estocásticos autorregresivos. En segundo lugar, con los modelos de redes neuronales artificiales LSTM, se evaluaron el desempeño de estos modelos de pronósticos mediante los índices de pronósticos. Finalmente, se eligió el mejor modelo para pronosticar los gastos medios diarios en la presa La Angostura.

4.1. PRONÓSTICO CON EL MEJOR MODELO ESTOCÁSTICO

La **Tabla 4.1** muestra los valores de cada índice de pronóstico para los mejores modelos estocásticos autorregresivos en la fase de validación de los datos, con menor $MAE(33.119 m^3/s)$, menor $MAPE(9.906 \%)$ y con un coeficiente de correlación de $R(0.919)$ entonces el modelo $ARIMA(3,1,3)$ es el mejor modelo estocástico autorregresivo para la predicción de caudal medio diario en la presa. La **Figura 4.1** muestra la relación entre el conjunto de datos de validación y el de predicción de cada uno de los tres modelos estocásticos autorregresivos.

Tabla 4.1. Índices de pronósticos de validación de los modelos estocásticos.

| MODELO | R | RMSE | MAE | MAPE [%] |
|----------------|-------|--------|--------|----------|
| $AR(4)$ | 0.917 | 72.733 | 34.005 | 14.056 |
| $ARMA(2,3)$ | 0.920 | 71.345 | 33.834 | 12.202 |
| $ARIMA(3,1,3)$ | 0.919 | 71.594 | 33.119 | 9.906 |

Considerando uno de los índices de pronósticos, el error porcentual medio absoluto $MAPE$ que es la escala para evaluar la exactitud del modelo usando el criterio del $MAPE$ determina que, un pronóstico muy exacto es el que tiene un valor menor o igual al 10%, un buen pronóstico tiene un valor entre un rango del 11% al 20%, un pronóstico razonable entre el 21% al 50% y un pronóstico inadecuado mayor al 50% (Lewis, 1982).

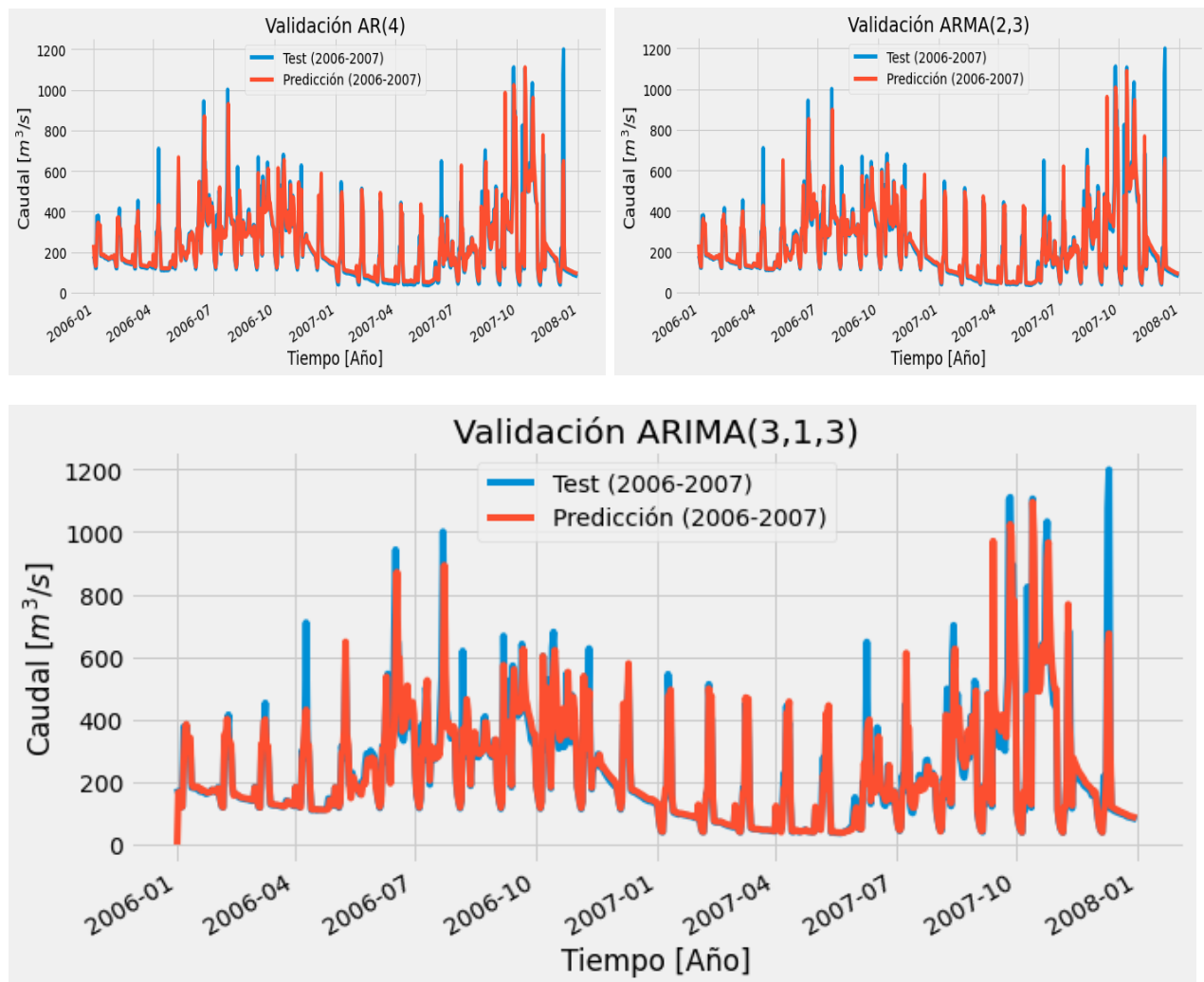


Figura 4.1. Pronósticos de la fase de validación de modelos estocásticos.

Entonces, el mejor modelo estocástico autorregresivo encontrado para la predicción de datos de validación es el modelo $ARIMA(3,1,3)$ ha arrojado un pronóstico exacto, ya que el valor de $MAPE$ se encuentra por debajo del 10%, el coeficiente de correlación indica que el modelo presenta un 91.9% de precisión del pronóstico en cuanto a la relación entre los caudales reales y pronosticados.

4.2. PRONÓSTICO CON EL MEJOR MODELO DE RED NEURONAL

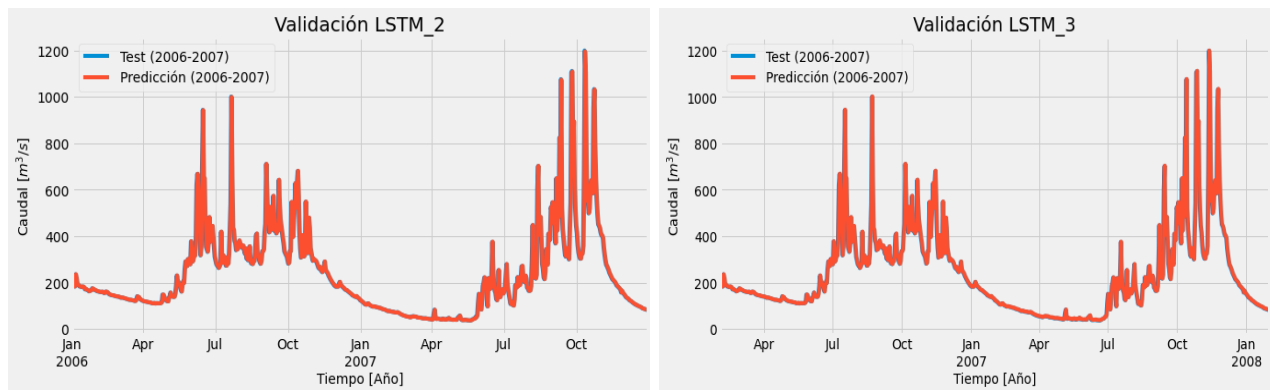
Se utilizaron los tres mejores modelos de red neuronal de la fase de entrenamiento para pronosticar los caudales del subconjunto de datos de la fase de validación del embalse. La **Tabla 4.2** muestra los valores de los índices de pronóstico para cada uno de los

modelos de redes neuronales artificiales recurrentes de memoria a corto plazo, con menor $RMSE(74.539 m^3/s)$, menor $MAE(30.390 m^3/s)$, menor $MAPE(8.279 \%)$ y un coeficiente de correlación de $R(0.912)$ entonces el modelo $LSTM_1$ es el mejor modelo de red neuronal artificial para la predicción de caudal medio diario en la presa, La Angostura. Y tomando como referencia el resultado del $MAPE$, el modelo arrojó un pronóstico exacto ya que el valor se encuentra por debajo del 10%, el coeficiente de correlación indica que el modelo presenta el 91.2% de precisión del pronóstico en cuanto a la relación entre los caudales reales y pronosticados.

Tabla 4.2. Índices de pronósticos de validación de los modelos de RNA de LSTM.

| MODELOS | R | RMSE | MAE | MAPE [%] |
|----------|-------|--------|--------|----------|
| $LSTM_1$ | 0.912 | 74.539 | 30.390 | 8.279 |
| $LSTM_2$ | 0.912 | 74.900 | 31.133 | 8.481 |
| $LSTM_3$ | 0.912 | 75.050 | 31.371 | 8.813 |

La **Figura 4.2** muestra la relación entre el subconjunto de datos de validación y su respectivo pronóstico de cada uno de los modelos de redes neuronales artificiales recurrentes de LSTM.



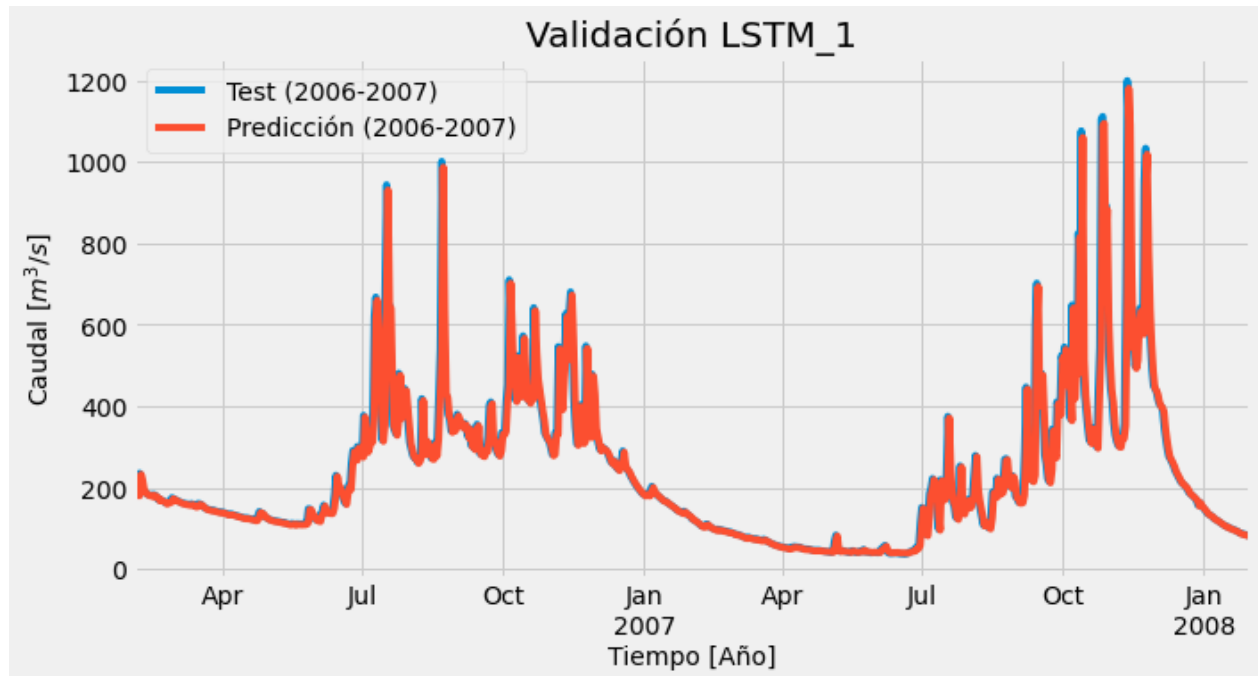


Figura 4.2. Pronósticos de la fase de validación de modelos RNA de LSTM.

4.3. COMPARACIÓN ENTRE LOS DOS MEJORES MODELOS DE ESTOCÁSTICO Y RED NEURONAL ARTIFICIAL

Finalmente, se hizo una comparación entre el mejor modelo estocástico autorregresivo y el mejor modelo de redes neuronales artificiales, a fin de decidir cuál es el mejor modelo para pronóstico de caudal medio diario en la presa. La **Tabla 4.3** muestra los índices de pronósticos de los dos mejores modelos de pronóstico en la fase de validación de los datos.

Tabla 4.3. Índices de comparación de los dos mejores modelos de pronóstico.

| MODELO | R | RMSE | MAE | MAPE [%] |
|---------------------|-------|--------|--------|----------|
| <i>ARIMA(3,1,3)</i> | 0.919 | 71.594 | 33.119 | 9.906 |
| <i>LSTM_1</i> | 0.912 | 74.539 | 30.390 | 8.279 |

Con menor *MAE*(30.390 m^3/s), menor *MAPE*(8.279 %) y un coeficiente de correlación de *R*(0.91) entonces el modelo *LSTM_1* es el mejor modelo para la predicción de caudal medio diario en la presa La Angostura.

Entonces, el mejor modelo encontrado para la predicción de datos de caudal medio diario en la presa La Angostura es el *LSTM_1*, que arrojó un pronóstico muy exacto, ya que el valor de *MAPE* se encuentra por debajo del 10%, el coeficiente de correlación indica que el modelo presenta un 91.2% de precisión del pronóstico en cuanto a la relación entre los caudales reales entrantes en la presa y los pronosticados.

La **Figura 4.3** muestra la ecuación de regresión lineal $y = 0.87x + 30.07$ del mejor modelo estocástico autorregresivo *ARIMA(3,1,3)* que tiene un coeficiente de correlación $R = 0.92$ y el mejor modelo de pronóstico de red neuronal *LSTM_1* tiene una ecuación de regresión lineal $y = 0.91x + 19.30$ y un coeficiente de correlación $R = 0.92$. Estas ecuaciones permitirán calcular valores de predicción de caudal entrante en la presa La Angostura para un día i sabiendo el caudal del día anterior $i - 1$, con un límite de confianza de 92% es decir el valor calculado podría ser más o menos 8% más bajo o alto.

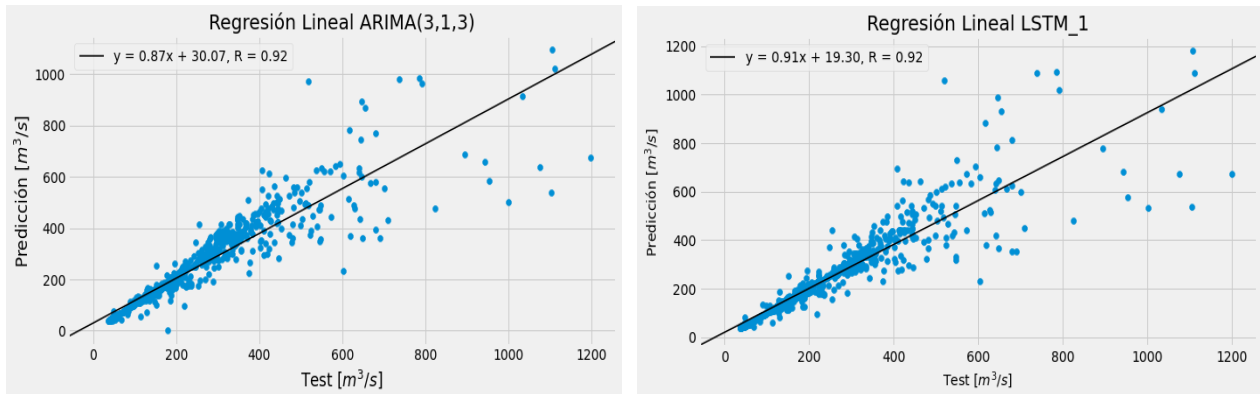


Figura 4.3. Ecuaciones de regresión lineal de una recta $y = ax + b$ de cada modelo.

CONCLUSIONES

Para la predicción de escurrimientos en embales existen diferentes modelos de predicción, que pueden ser físicos o empíricos. La elección del modelo de predicción adecuado depende de la disponibilidad de los datos, la complejidad de la cuenca hidrográfica, los objetivos y la finalidad de la predicción.

El modelo de las redes neuronales artificiales son una herramienta útil para la predicción de escurrimiento en presas, pueden ayudar en la toma de decisiones sobre la gestión del recurso hídrico y prevención de posibles inundaciones.

Los modelos de red neuronal artificial son un conjunto de algoritmos de aprendizaje automático que imitan la estructura y el comportamiento de las redes neuronales biológicas, se componen de múltiples capas de neuronas artificiales que permiten procesar los datos y ajustar los pesos de las conexiones para mejorar la precisión de la predicción. Estos modelos son más adecuados para resolver problemas más complejos que involucran patrones no lineales y relaciones multivariadas.

Por otro lado, los modelos autorregresivos son modelos estocásticos que se pueden utilizar para la predicción de valores en una serie temporal en función de los datos anteriores de la misma serie. Estos modelos utilizan una relación lineal para predecir datos que siguen patrones estacionales o tendencias, son simples y efectivos para datos con patrones claros y repetitivos.

Los modelos autorregresivos y los modelos de redes neuronales artificiales son dos enfoques comunes que se utilizan para modelar series temporales y predecir su comportamiento al futuro.

Este trabajo de investigación consistió en la comparación de modelos estocásticos $AR(p)$, $ARMA(p, q)$ y $ARIMA(p, d, q)$ contra modelos de redes neuronales artificiales recurrentes de memoria a corto plazo $LSTM$ para predicción de escurrimientos medio diario en la presa La angostura, Chiapas, México. Con la finalidad de tener

herramientas de predicción más confiables y útiles para un buen manejo y aprovechamiento de los recursos hídricos de la presa.

Primero en la comparación de los resultados de predicción de gastos medios diarios de los modelos estocásticos autorregresivos $AR(4)$ y $ARMA(2,3)$ dieron valores de $MAPE = 14.06\%$ y $MAPE = 12.20\%$ respectivamente son superiores a 10% , los modelos han arrojado pronósticos razonables. Al momento de considerar los parámetros de media móvil y la diferenciada de los datos para que sean estacionarios se mejoró bastante los valores de pronósticos con el modelo $ARIMA(3,1,3)$ dio un valor de $MAPE = 9.91\%$ que indica que cada valor pronosticado pudiera tener en promedio un error más o menos de 9.91% que arroja un buen pronóstico.

Pero la relación que existe entre cada valor pronosticado y los datos anteriores utilizados para el pronóstico no tiene patrones muy claros o repetitivos entonces los valores del error porcentual absoluto medio han sido muy altos para los modelos estocásticos autorregresivos, debido a que los modelos autorregresivos suponen que los valores futuros dependen únicamente de los valores pasados. Y con los cambios abruptos en las series temporales debido a los eventos de inundaciones (Tormentas Javier, Matthew y huracán Stan) que fueron en septiembre y octubre de 2004, esas lluvias torrenciales han modificado el patrón de las series temporales y han perjudicado a los modelos auto regresivos en la fase de entrenamiento y validación de los datos.

Aunque el $ARIMA(3,1,3)$ es el mejor modelo estocástico autorregresivo e hizo que la serie sea estacionaria para el conjunto de datos de entrenamiento y validación, pero la red neuronal artificial recurrente $LSTM$ utilizado en este trabajo de investigación es superior para la predicción de escurrimiento en la presa dio un valor de $MAPE = 8.279\%$, que significa el pronóstico tiene un error más o menos en promedio de 8.28% muy inferior respecto al valor del error del mejor modelo estocástico autorregresivo. Este modelo de red neuronal tiene una configuración de hiperparámetros de una capa oculta que contiene 32 neuronas, una tasa de aprendizaje 0.1% , tamaño de lote de datos igual a 30; 2 000 épocas y la función de activación tangente hiperbólica, es el mejor modelo para pronóstico de escurrimiento medio diario en la presa La Angostura. Una de las

desventajas de las redes neuronales artificiales es que no son transparentes; es decir, pueden ser muy complejas, lo que dificulta entender cómo se está tomando una decisión. Esto puede ser un problema si se necesita explicar el razonamiento detrás de un pronóstico porque involucra a una función que se usa como caja negra y se les aclara que no es el alcance de trabajo de investigación.

Esta metodología aplicada en este trabajo de investigación se puede replicar en otras zonas de estudio con las características similares y disponibilidad de información sobre estaciones hidroclimatológicas para predicción de precipitaciones, temperaturas, caudales, escurrimientos, manejo y operación de presas para generación de energía hidroeléctrica y también para la prevención y mitigación de inundaciones en zonas ubicadas aguas abajo de una presa. Y los resultados obtenidos muestran que los modelos estocásticos autorregresivos pueden ser una herramienta de predicción con muy buenos resultados mientras que los datos de las series temporales presenten patrones claros o repetitivos, sin embargo, si los datos tienen una relación no lineal es mejor el uso de las redes neuronales artificiales porque son capaces de interpretar patrones de muy alta complejidad.

RECOMENDACIONES

En este trabajo de investigación solamente se consideró como datos de entrada los valores de caudales medios diarios para los modelos estocásticos autorregresivos y los modelos de redes neuronales artificiales.

Como temas para otros trabajos de investigación podrían simular otros modelos considerando aparte de los valores de caudales también los datos de precipitación y temperatura en la zona de estudio; y también infiltración y evaporación en el embalse de la presa La Angostura para comparar los resultados con el nivel medio diario de agua registrado en la presa.

Otro tema podría ser también la búsqueda de la representación analítica de la función que utiliza las redes neuronales artificiales para entrenamiento y validación de datos.

REFERENCIAS

Alegre, A. (2003). *Simulación de redes neuronales artificiales: Una aplicación didáctica*. Tesis Lic. Sistemas. Universidad Nacional del Nordeste. Corrientes, Argentina. 188 p.

Bergstra, J., Bengio, Y. (2012). *Random Search for Hyper-Parameter Optimization*. Journal of Machine Learning Research 13 (2012) 281-305. Consultado el 02 de febrero de 2023 en: <https://jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf>

Box, G.E., Jenkins, G.M. (1970). *Time series analysis forecasting and control*. Holden day series in time series analysis, San Francisco. USA.

Box, G.E., Müller, M.E. (1958). *A note on the generation of normal deviates*. Annals of Mathematics Statistics. 28: 610-611.

CONAGUA (Comisión Nacional del Agua). (2010). *Estadísticas del agua en México*. Edición 2010, México, Secretaría del Medio Ambiente y Recursos Naturales, 2011, p. 83

CONAGUA (Comisión Nacional del Agua). (2018). *Estadísticas del Agua en México*. Edición 2018. Ciudad de México, México.

Dölling, O.R., Varas, C.E. (2000). *Pronóstico de Caudales en Cuencas Nivales Usando Redes Neuronales Artificiales*. XIX Congreso Latinoamericano de Hidráulica. Cordoba, Arg.: IAHS.

Domínguez, M.R., Cruickshank, C., Arganis, J.M., Rodríguez, K. (2004). *Obtención de curvas de pronóstico de escurrimiento directo tomando en cuenta datos de precipitación efectiva usando programación genética*. Instituto de Ingeniería, UNAM.

Eckstein, D., Kunzel, V., Schafer, L. (2021). *IRC (Índice de Riesgo Climático), Resumen del Índice de riesgo climático global*. Germanwatch. Retomado de la siguiente liga: https://www.germanwatch.org/sites/default/files/Resumen%20Indice%20de%20Riesgo%20Clim%C3%A1tico%20Global%202021_0.pdf

Escalante, S.C., Reyes, C.L. (2002). *Técnicas estadísticas en hidrología*. México, UNAM. Facultad de Ingeniería.

Freeman, J.A., Skapura, D.M. (1991). *Neural networks: algorithms, applications and programming techniques*. (Computation and Neural Systems Series) 401 p. Addison-Wesley Pub. Co., Reading, Massachusetts, USA.

Hecht-Nielsen, R. (1987). *Neurocomputing: Picking the human brain*. IEEE spectrum, 25, pág. 36-41, marzo 1988. Reimpreso en el texto: Artificial Networks: theoretical Concepts. V. Venturi Ed. Pág. 13-18, IEEE Computer Society Press Technology Series.

Hilera, J.R., Martínez, V.J. (1995). *Redes neuronales artificiales: fundamentos, modelos y aplicaciones*. Editorial Alfaomega, Madrid, España.

Kohonen, T. (1988). *An introduction to neural computing*. Neural networks. Vol. 1 pág. 3-16.

Lewis, C.D. (1982). *Industrial and business forecasting methods: A practical guide to exponential smoothing and curve fitting*. Butterworth Heinemann.

Maren, A.J., Harston, C.T. and Pap, R.M. (1990). *Handbook of Neural Computing Applications*. Ed. Academic Press.

Matplotlib Comunity. (2022). *Matplotlib*. <https://matplotlib.org/stable/index.html>

Meneses-Bautista, F.D., Matias, A. (2017). *Pronóstico del tipo de cambio USD/MXN con redes neuronales de retropropagación*. Research in Computing Science 139, ISSN 1870-4069

Numpy Comunity. (2022). *NumPy Reference. Release 1.23.0*. Written by the NumPy community. <https://numpy.org/doc/>

Pandas Comunity. (2022). *Pandas*. <https://pandas.pydata.org/docs/>

Rosenblatt, F. (1958). *The Perceptron: A probabilistic model for information storage and organization in the brain*. Psychological Review, 65, pp. 386-408, 1958. Reimpreso en el texto "Neurocomputing" (J. Anderson y E. Rosenfeld ed.), pp. 92-114, MIT Press, 1988.

Russell, S.J., Norvig, P. (2004). *Inteligencia artificial. Un enfoque moderno*. Pearson Educación (Segunda edición) (Madrid), ISBN 978-84-205-4003-0.

Shamseldin, A.Y., O'Connor, K.M., Liang, G.C. (1997). *Methods for combining the output of different rainfall runoff models*. J. Hydrol., 197, 203–229.

Solomatine, D. (2002). *Data-driven modelling: machine learning and data mining in water related problems*. Tutorial handouts. V International Conference on Hydroinformatics. Cardiff, UK. 1-5 July.

Theckel, J.T., Rana, S., Gupta, S., Venkatesh, S. (2016). *Hyperparameter Tuning for Big Data using Bayesian Optimisation*. 2016 23rd International Conference on Pattern Recognition (ICPR).

McCulloch, S.W., Pitts, W. (1943). *A logical calculus of the ideas immanent in nervous activity*. Bulletin of Mathematical Biology Vol. 52, No. 1/2. pp. 99-115. 1990. Printed in Great Britain.

Torres, J. (2020). *Python deep learning. Introducción practica con keras y tensorflow 2*. ISBN: 978-84-267-2828-9. Edición Alfaomega. Barcelona, España.

Viñuela, P.I., Galván, L.I. (2004). *Redes neuronales artificiales. Un enfoque practico*. Editorial Pearson Prentice Hall, Madrid, España.

Widrow, B., Lehr, M.A. (1959). *30 Years of Adaptive Neural Networks: Perceptron, Madaline and Backpropagation*. Proceedings of the IEEE, Vol.78, No.9, pp. 1415-1442, septiembre, 1990.