

**UNIVERSIDAD NACIONAL AGRARIA
LA MOLINA**

**ESCUELA DE POSGRADO
MAESTRÍA EN RECURSOS HÍDRICOS**



**“SIMULACIÓN Y PRONÓSTICO DE CAUDALES DIARIOS
DEL RÍO AMAZONAS USANDO UN ENFOQUE HÍBRIDO
WAVELET Y REDES NEURONALES”**

Presentada por:

LUCIO EUSEBIO VERGARA SATURNO

**TESIS PARA OPTAR EL GRADO DE
MAGISTER SCIENTIAE EN RECURSOS HÍDRICOS**

Lima – Perú

2020

**UNIVERSIDAD NACIONAL AGRARIA
LA MOLINA**

**ESCUELA DE POSGRADO
MAESTRÍA EN RECURSOS HÍDRICOS**

**“SIMULACIÓN Y PRONÓSTICO DE CAUDALES DIARIOS
DEL RÍO AMAZONAS USANDO UN ENFOQUE HÍBRIDO
WAVELET Y REDES NEURONALES”**

**TESIS PARA OPTAR EL GRADO DE
*MAGISTER SCIENTIAE***

Presentada por:

LUCIO EUSEBIO VERGARA SATURNO

Sustentada y aprobada ante el siguiente jurado:

Mg.Sc. Toribio S. Santayana Vela
PRESIDENTE

Dr. Waldo S. Lavado Casimiro
PATROCINADOR

Ph.D. Eduardo A. Chávarri Velarde
MIEMBRO

Mg.Sc. Miguel Sánchez Delgado
MIEMBRO

DEDICATORIA

A mi madre Florlinda Enma, por lo que siento un profundo amor, respeto y orgullo, porque siempre ha sido fuente de inspiración y ejemplo, gracias a ella es que soy como soy.

A la memoria de mi hermano, Edgar Ramiro, quien me inspiró todo el tiempo para ser un mejor hombre.

A mi hermano Rulem Chale, con quien compartimos buenos momentos de nuestra infancia y juventud.

A mi prometida Ofelia, que tuvo la paciencia suficiente para apoyarme profundamente, por darme su comprensión, su cariño y su amor.

A las personas que me han apoyado y estuvieron a mi lado durante el desarrollo de esta tesis.

AGRADECIMIENTOS

Me gustaría aprovechar esta oportunidad para expresar mi sincera gratitud a mi patrocinador, el Dr. Waldo Lavado, cuyo constante aliento, dirección y comentarios oportunos me ayudaron a terminar esta tesis. Mi agradecimiento también se extiende al Dr. Jhan Carlo Espinoza, por su apoyo y orientación con sus valiosos aportes, motivación y críticas a lo largo de este proceso de investigación.

Deseo agradecer a los miembros de mi comité consejero, que fueron más que generosos con su experiencia y su valioso tiempo. Estoy agradecido con el Mg.Sc. Toribio Santayana y el Mg.Sc. Miguel Sánchez, por sus críticas constructivas para mi investigación. Especialmente, agradezco los útiles comentarios y la orientación del Ph.D. Eduardo Chávarri, que ha llevado a una serie de mejoras.

Además, me gustaría agradecer a los responsables del convenio de cooperación interinstitucional entre la Autoridad Nacional del Agua y el Instituto Geofísico del Perú, por otorgarme la oportunidad de realizar el presente trabajo de investigación y por sus generosidades al haberme facilitado siempre los medios suficientes para llevar a cabo todas las actividades durante el desarrollo de esta tesis.

Por último, pero no menos importante, me gustaría agradecer a los miembros de mi familia, especialmente a mi madre Florlinda y mi hermano Rulem, por su estímulo permanente, comprensión y apoyo generoso en todas las circunstancias. Me gustaría extender mi agradecimiento especial a mi adorable prometida, la Ing. Ofelia Rojas, por su apoyo continuo, sus valiosos comentarios y la edición de esta tesis.

Sobre todo, me faltan palabras para agradecer al Dios Todopoderoso porque me ha dado resistencia, salud y capacidad para completar con éxito mi trabajo de tesis.

A todas las personas que de alguna manera han contribuido a esta investigación...

¡Gracias!

ÍNDICE GENERAL

DEDICATORIA	i
AGRADECIMIENTOS	ii
ÍNDICE GENERAL	iii
ÍNDICE DE TABLAS	viii
ÍNDICE DE FIGURAS	ix
ÍNDICE DE ANEXOS	xii
RESUMEN	xiii
ABSTRACT	xiv
I. INTRODUCCIÓN	1
1.1. Situación problemática	1
1.2. Justificación de la investigación	2
1.3. Alcances y limitaciones	6
1.4. Objetivos	8
1.4.1. Objetivo general	8
1.4.2. Objetivos específicos	8
II. REVISIÓN DE LITERATURA	9
2.1. Conceptos básicos	9
2.1.1. Descripción y análisis preliminar	9
2.1.2. Definición de modelamiento y simulación	10
a. Modelamiento.....	10
b. Simulación.....	10
2.1.3. Pronóstico hidrológico	11
a. Métodos de pronóstico	11
b. Categorías de pronóstico hidrológico.....	13

c. Horizontes de pronóstico.....	14
2.1.4. Control de un proceso	16
2.2. Redes neuronales artificiales.....	16
2.2.1. Definición	17
2.2.2. Breve reseña histórica de las redes neuronales	17
2.2.3. La neurona biológica	18
2.2.4. La neurona artificial	19
2.2.5. Elementos básicos de una red neuronal	20
2.2.6. Funciones de activación más populares	21
2.2.7. Clasificación de las RNA	22
a. Conexiones entre neuronas (topología)	22
b. Mecanismos de aprendizaje	24
c. Tipo de neuronas	26
2.2.8. Tipos de redes neuronales artificiales	27
a. <i>Perceptrón</i> simple	27
b. <i>Perceptrón</i> multicapa (MLP)	29
c. Redes de funciones de base radial (RBF)	30
2.2.9. <i>Perceptrón</i> multicapa	31
2.2.10. Algoritmos para la optimización	34
a. Algoritmo de <i>Levenberg-Marquardt</i>	35
b. Algoritmo de regla delta generalizada	36
c. Algoritmo de <i>Resilient Backpropagation</i>	37
2.2.11. Aprendizaje y generalización de las RNA	37
2.2.12. Ventajas y limitaciones de las RNA	38
2.2.13. Pronóstico de series de tiempo utilizando RNA	39
a. Predictor individual	40
b. Series de predictores múltiples	41

2.2.14. Aplicaciones de RNA para el pronóstico de caudales en el Perú	42
2.3. Descripción de la transformada de <i>wavelet</i>	43
2.3.1. Introducción a las transformadas	43
2.3.2. La transformada de <i>Fourier</i>	44
2.3.3. Transformada de <i>wavelet</i>	45
a. Transformada de <i>wavelet</i> continua.....	46
b. Transformada de <i>wavelet</i> discreta	48
2.3.4. Familia de funciones <i>wavelet</i>	48
2.3.5. Análisis multiresolución.....	51
a. Filtros de un nivel.....	51
b. Filtros multinivel	52
2.3.6. Determinación del número de niveles.....	53
2.4. Enfoques de modelamiento híbrido.....	53
2.4.1. Introducción a las técnicas híbridas	54
2.4.2. Enfoques de la inteligencia artificial.....	55
2.4.3. Aplicaciones de modelos híbridos <i>wavelet</i> - IA.....	58
2.4.4. Directrices generales para los modelos híbridos.....	61
a. Selección del tipo de transformada de <i>wavelet</i>	61
b. Selección de la función <i>wavelet</i>	62
c. Selección del nivel de descomposición	62
d. Selección del modelo “basados en datos”	63
e. Identificación de subseries <i>wavelet</i> dominantes.....	64
III. MATERIALES Y MÉTODOS.....	66
3.1. Descripción del ámbito de estudio	66
3.1.1. Estación hidrométrica de Tamshiyacu	67
3.2. Información disponible y recursos	68
3.3. Análisis preliminar de datos	69

3.4. Desarrollo de modelos de redes neuronales artificiales.....	69
3.4.1. Protocolo para la implementación de las RNA.....	70
a. División de la data.....	72
b. Arquitectura del tipo <i>perceptrón</i> multicapa.....	73
c. Preprocesamiento de datos.....	75
d. Selección de entradas relevantes.....	76
e. Selección de neuronas de la capa oculta.....	77
f. Criterios de parada de entrenamiento.....	78
g. Entrenamiento de la red neuronal.....	79
h. Optimización del MLP.....	80
i. Validación o prueba de los modelos.....	81
3.4.2. Medidas estadísticas de desempeño.....	81
3.4.3. Implementación de las RNA en <i>Matlab</i>	83
3.5. Desarrollo de modelos híbrido <i>wavelet</i> redes neuronales.....	84
3.5.1. Submodelo <i>wavelet</i>	86
3.5.2. Submodelo redes neuronales.....	86
3.5.3. Directrices para los modelos híbridos WRN.....	88
a. Selección del tipo de transformada de <i>wavelet</i>	89
b. Selección de la función <i>wavelet</i>	89
c. Selección del nivel de descomposición.....	89
d. Selección del modelo “basados en datos”.....	90
e. Identificación de subseries <i>wavelet</i> dominantes.....	90
3.5.4. Implementación del análisis <i>wavelet</i> en <i>Matlab</i>	91
3.6. Comparación de las dos técnicas de la IA.....	91
3.6.1. Pronósticos a un día de anticipación.....	92
3.6.2. Pronósticos con horizontes más amplios.....	93

IV. RESULTADOS Y DISCUSIÓN	94
4.1. Análisis preliminar de datos	95
4.1.1. Estadísticos básicos y patrones predominantes.....	95
a. Tendencia	95
b. Estacionareidad.....	96
c. Variabilidad del régimen de caudales.....	98
4.2. Modelos basados en redes neuronales artificiales.....	99
4.2.1. Segmentación y preprocesamiento de datos	100
4.2.2. Patrones de entrada a las RNA	100
4.2.3. Modelos de RNA para pronóstico a un día de anticipación	102
a. Comparación del desempeño de las RNA	106
b. Modelo seleccionado de RNA	107
4.2.4. Ampliación del horizonte de pronóstico	108
4.3. Modelos híbridos <i>wavelet</i> redes neuronales.....	111
4.3.1. Análisis de descomposición <i>wavelet</i>	112
4.3.2. Patrones de entrada a los modelos híbridos WRN	114
4.3.3. Modelos de WRN para pronóstico a un día	115
a. Comparación de modelos WRN.....	118
b. Modelo seleccionado de WRN.....	119
4.3.4. Ampliación del horizonte de pronóstico	120
4.4. Comparación de las dos técnicas de pronóstico	124
4.4.1. Pronósticos a un día de anticipación	124
4.4.2. Pronósticos con horizontes más amplios.....	125
V. CONCLUSIONES.....	129
VI. RECOMENDACIONES.....	132
VII. REFERENCIAS BIBLIOGRÁFICAS.....	133
VIII. ANEXOS.....	147

ÍNDICE DE TABLAS

Tabla 1: Resumen de estadísticos básicos de la serie de caudales diarios total, subseries de entrenamiento y validación del río Amazonas.....	96
Tabla 2: Patrones de entrada y salida de los modelos RNA para pronósticos de caudales a un día de anticipación.....	102
Tabla 3: Resumen de desempeño de los modelos RNA para pronósticos a un día de anticipación, en la fase de entrenamiento y validación.....	104
Tabla 4: Patrones de salida del modelo RNA4 para pronósticos de caudales a múltiples días de anticipación.....	108
Tabla 5: Resumen de desempeño del modelo RNA4 para múltiples horizontes de pronóstico, en la fase de entrenamiento y validación.....	109
Tabla 6: Patrones de entrada y salida de los modelos híbridos WRN para el pronóstico de caudales diarios a un día anticipación.....	114
Tabla 7: Resumen de desempeño de los modelos híbridos WRN para pronósticos a un día de anticipación, en la fase de entrenamiento y validación.....	116
Tabla 8: Patrones de salida del modelo híbrido WRN12 para pronósticos de caudales a múltiples días de anticipación.....	121
Tabla 9: Resumen de desempeño del modelo WRN12 para pronósticos a múltiples días de anticipación, en la fase de entrenamiento y validación.....	121
Tabla 10: Por ciento de aumento y disminución de los índices de desempeño del modelo híbrido WRN en comparación al modelo RNA para pronóstico a un día de anticipación, en la fase de entrenamiento y validación.....	125
Tabla 11: Comparación de desempeño de las dos técnicas de pronóstico, para múltiples horizontes de pronóstico, en la fase de entrenamiento.....	126
Tabla 12: Comparación de desempeño de las dos técnicas de pronóstico, para múltiples horizontes de pronóstico, en la fase de validación.....	127

ÍNDICE DE FIGURAS

Figura 1: Inundaciones históricas en la ciudad peruana de Iquitos en el 2012	2
Figura 2: Tipos básicos de métodos de pronóstico	12
Figura 3: Principales categorías de pronóstico hidrológico	14
Figura 4: Horizontes de pronóstico y método básicos de pronóstico.....	15
Figura 5: Componentes de tiempo y espacio del “tiempo de anticipación”.....	16
Figura 6: Modelo simplificado de una neurona biológica	18
Figura 7: Modelo de una neurona artificial.....	20
Figura 8: Elementos de una red neurona artificial (RNA).....	20
Figura 9: RNA alimentada hacia adelante.....	22
Figura 10: RNA con retroalimentación.....	23
Figura 11: Aprendizaje supervisado.....	24
Figura 12: Aprendizaje autoorganizado	25
Figura 13: <i>Perceptrón</i> simple.....	28
Figura 14: <i>Perceptrón</i> multicapa (MLP).....	29
Figura 15: Red de función de base radial (RBF).....	31
Figura 16: Sobreajuste o sobreaprendizaje (<i>overfitting</i>) en una red neuronal.....	38
Figura 17: Modelo de pronóstico de series de tiempo univariado	40
Figura 18: Modelo de pronóstico de series de tiempo multivariado	41
Figura 19: Esquema de transformación de la transformada de <i>Fourier</i>	44
Figura 20: Esquema de transformación de la transformada de <i>wavelet</i>	46
Figura 21: Ilustración de <i>wavelet</i> (a) desplazado y (b) escalado.....	47
Figura 22: Familia de <i>wavelet Daubechies</i>	49
Figura 23: Función <i>wavelet Haar</i>	50

Figura 24: Funciones <i>wavelet Mexican-Hat, Morlet, Coiflet1</i> y <i>Symlet2</i>	50
Figura 25: Diagrama de descomposición de señales usando bancos de filtros	51
Figura 26: Descomposición de señales usando bancos de filtros y submuestreo	52
Figura 27: Árbol de descomposición <i>wavelet</i> (descomposición múltiple)	53
Figura 28: Esquema del modelo de pronóstico híbrido <i>wavelet-IA</i>	57
Figura 29: Ubicación de Perú en América del sur, cuenca del Amazonas en América del sur y países involucrados, estación hidrométrica de Tamshiyacu dentro de la amazonía peruana.....	67
Figura 30: Principales pasos en el desarrollo de modelos de RNA.....	71
Figura 31: Resumen Esquema de un <i>perceptrón</i> multicapa con una capa oculta.....	74
Figura 32: Estructura del modelo híbrido <i>wavelet</i> red neuronal (WRN) para pronósticos N pasos adelante.....	85
Figura 33: Principales pasos en el desarrollo de los modelos híbridos WRN.....	88
Figura 34: Serie temporal de caudales diarios del río Amazonas (Tamshiyacu) de 01 de setiembre de 1985 a 31 de agosto de 2012.....	96
Figura 35: Gráfico de la FAC y FACP de la serie de caudales de 1985 al 2012 (superior), detalle de la FAC para los primeros 100 retardos (inferior).....	97
Figura 36: Variación del régimen de caudales mensuales en la estación hidrométrica Tamshiyacu.....	99
Figura 37: Serie de caudal diario de registro histórico original (izquierda) y escalda en el rango $[-1,1]$ (derecha).....	100
Figura 38: Modelo RNA4 (4-N-1) con cuatro entradas, N nodos ocultos y una salida.....	101
Figura 39: Serie de errores residuales del modelo RNA4 en la fase de calibración y validación. Función de autocorrelación, histograma y periodograma acumulada en la fase de validación.....	105
Figura 40: Desempeño del modelo RNA4 para múltiples horizontes de pronóstico en la fase de validación.....	110

Figura 41: Hidrograma de caudales observados y pronosticados por el modelo RNA4 para 21 días de anticipación, en la fase de calibración y validación. Detalle y gráfico de dispersión de la fase de validación.....	111
Figura 42: Selección del orden de la familia <i>wavelet Daubechies</i> en función al índice RMSE del modelo RNA1 con tres niveles de descomposición en la fase de entrenamiento, para las 30 primeras neuronas ocultas.....	112
Figura 43: Descomposición <i>wavelet</i> con cuatro niveles de descomposición utilizando <i>wavelet</i> madre <i>Daubichies</i> de orden 2 (db2).....	113
Figura 44: Serie de errores residuales del modelo WRN12 en la fase de calibración y validación. Función de autocorrelación, histograma y periodograma acumulada en la fase de validación.....	117
Figura 45: Desempeño del modelo híbrido WRN12 para múltiples horizontes de pronóstico en la fase de validación.....	122
Figura 46: Hidrograma de caudales observados y pronosticados por el modelo WRN12 para 30 días de anticipación, en la fase de calibración y validación. Detalle y gráfico de dispersión de la fase de validación.....	123
Figura 47: Comparación gráfica de performance del modelo RNA4 y el modelo híbrido WRN12 para múltiples horizontes de pronóstico en la fase de validación, a través de los estadísticos RMSE y NSE.....	128

ÍNDICE DE ANEXOS

ANEXO 1: Generalidades del <i>software Matlab</i>	147
ANEXO 2: Metodología de redes neuronales artificiales en <i>Matlab</i>.....	148
ANEXO 3: Metodología de análisis <i>wavelet</i> mediante <i>Matlab</i>.....	154
ANEXO 4: Código <i>Matlab</i> para la aplicación de modelos RNA	161
ANEXO 5: Código <i>Matlab</i> para la aplicación de modelos híbridos WRN.....	163

RESUMEN

El incremento de eventos extremos durante las últimas décadas en la cuenca amazónica, ha dado lugar a un creciente interés por implementar efectivos sistemas de pronóstico hidrológico. Los pronósticos a corto plazo, como parte intrínseca de estos sistemas, son fundamentales en la mitigación de inundaciones, y la gestión de los recursos hídricos. Debido a la importancia de los pronósticos de alta calidad y a la complejidad de los sistemas hidrológicos, se han estudiado un gran número de métodos de modelamiento orientado a pronósticos. En esta investigación, se desarrollaron modelos “basados en datos” con dos técnicas, la red neuronal artificial (RNA) y un enfoque híbrido que combina análisis multiresolución *wavelet* y RNA llamado modelo *wavelet* red neuronal (WRN). En efecto, se formularon distintas estructuras de modelos univariados de RNA y WRN para múltiples horizontes de pronóstico, considerando que la confiabilidad de pronóstico disminuye al aumentar el tiempo de anticipación. Para el cual, se empleó series observadas de caudales diarios para el periodo 1985-2012, registrados en la estación hidrológica de Tamshiyacu en el río Amazonas, Perú. Además, el desempeño de los modelos se evaluó en función a los índices estadísticos, tales como la raíz del error cuadrático medio (RMSE) y la eficiencia de *Nash-Sutcliffe* (NSE). Así, para el horizonte de pronóstico más lejano (30 días), se encontró que el modelo WRN con $RMSE = 4820 \text{ m}^3/\text{s}$ y $NSE = 0.83$ superó ampliamente al modelo RNA con $RMSE = 6092 \text{ m}^3/\text{s}$ y $NSE = 0.72$, en la etapa de validación. Estos hallazgos muestran que el modelo híbrido tiene la capacidad potencial para mejorar la precisión de pronóstico en comparación al modelo RNA convencional. En suma, los resultados de esta investigación ayudarán a los hidrólogos y tomadores de decisiones en el pronóstico de caudales y la gestión sostenible de los recursos hídricos.

Palabras clave: Pronóstico a corto plazo; red neuronal artificial; análisis multiresolución *wavelet*; modelo híbrido; río Amazonas.

ABSTRACT

The increasing number of extreme events during the last decades in the Amazon basin has led to a growing interest in implementing effective hydrological forecasting systems. Short-term forecasts, as an intrinsic part of these systems, are crucial for flood mitigation and water resources management. Due to the importance of high-quality forecasting and the complexity of hydrological systems, a large number of forecasting-oriented modelling methods have been studied. In this research, data-driven models with two techniques were developed, artificial neural network (ANN) and a hybrid approach which combines wavelet multi-resolution analysis and ANN named wavelet neural network (WNN) model. In effect, several structures of univariate ANN and WNN models were formulated for multiple forecasting horizons, considering that the reliability of forecasting decreases with increasing the lead-time. For which, observed time series of daily streamflows for the period 1985-2012 recorded at the Tamshiyacu gauging station on the Amazon river, Peru, were used. In addition, the performance of the models has been evaluated based on the statistical indices, such as root mean square error (RMSE) and Nash-Sutcliffe efficiency (NSE). Thus, for longer lead-time forecasting (30 days), it was found that the WNN model with $RMSE = 4820 \text{ m}^3/\text{s}$ and $NSE = 0.83$, widely outperformed to ANN model with $RMSE = 6092 \text{ m}^3/\text{s}$ and $NSE = 0.72$, in the test period. These findings show that the hybrid WNN model has the potential ability to improve the forecasting accuracy compared to the conventional ANN model. In sum, the outcomes of this research will assist hydrologists and decision makers in streamflow forecasting and sustainable management of water resources.

Keywords: Short-term forecasting; artificial neural network; wavelet multi-resolution analysis; hybrid model; Amazon River.

I. INTRODUCCIÓN

1.1. SITUACIÓN PROBLEMÁTICA

El Perú cubre cerca del 16 por ciento del total de la cuenca del Amazonas, que a su vez representa más del 76 por ciento (977 920 km²) del territorio nacional, de ahí que es considerado un país amazónico, además, el 98 por ciento de sus recursos hídricos se encuentran en dicha región (DGAS 1995). Recientes estudios científicos muestran que, durante los últimos cuarenta años, la parte oeste de la cuenca amazónica (Amazonas peruano) ha sufrido severos eventos hidrológicos extremos, como sequías agudas en 1995, 1998, 2005 y 2010, así como fuertes inundaciones en 1993, 1999, 2009 y 2012 (Espinoza *et al.* 2009a; Lavado *et al.* 2012; Espinoza *et al.* 2011a; Espinoza *et al.* 2012; Espinoza *et al.* 2013). Los cuales resultan muy perjudiciales para las poblaciones cercanas a los cursos de los ríos, así como para la agricultura y los ecosistemas amazónicos. Por ejemplo, la sequía del 2010, produjo un elevado incremento de los precios de los alimentos de primera necesidad en Iquitos debido a problemas en el transporte fluvial. Por su parte, la fuerte crecida del 2012 (Figura 1) ocasionó una severa inundación en Iquitos y las principales ciudades de Loreto, estimándose en más de 150 mil damnificados en la región. Además, han evidenciado un incremento de estos eventos extremos, particularmente en los países andino-amazónicos (Perú, Bolivia, Ecuador y Colombia) desde inicios de los años 1990 (Espinoza *et al.* 2009a).

Estas graves sequías e inundaciones extremas en la hidrología de la amazonía peruana plantean retos importantes para la planificación y operación de los mismos. Por lo tanto, en el intento de hacer frente a estos eventos anómalos cada vez más frecuentes, se requieren de efectivos sistemas de pronóstico hidrológico (SPH) para monitorear, analizar, evaluar y pronosticar de forma precisa las condiciones hidrológicas en esta región. Ya que, estos SPH son importantes para reducir riesgos hidrológicos y son útiles en la planificación de la toma de decisiones para reducir damnificados; además, el uso de los SPH se enfatiza aún más con la mejora de la comprensión de los procesos del clima y los posibles cambios climáticos. Asimismo, se debe tener en cuenta que un SPH bien establecido orientado a

inundaciones, que atiende operaciones de prevención de desastres, debería en la mayoría de los casos, ser más eficientes que las medidas estructurales, en la mitigación de los efectos de inundaciones mayores muy poco frecuentes (Nemec 1986; Mashor 2002). En los últimos años, se ha dedicado un esfuerzo considerable al problema del pronóstico hidrológico en tiempo real. Así, los países desarrollados se ocupan, principalmente, de la expansión y mejora de los servicios existentes de pronóstico de caudales en los ríos para diferentes fines. Sin embargo, los países en desarrollo como el Perú con apoyo internacional o bilateral aún están concentrados en el establecimiento de redes hidrológicas básicas y una capacidad inicial de pronóstico hidrológico, principalmente, para evitar la pérdida de vidas y daños materiales como consecuencia de las inundaciones y las sequías (Nemec 1986).



Figura 1: Inundaciones históricas en la ciudad peruana de Iquitos en el 2012
FUENTE: IRD SO HYBAM: <http://www.ore-hybam.org/>

1.2. JUSTIFICACIÓN DE LA INVESTIGACIÓN

Los pronósticos hidrológicos como proveedores de información a los SPH, y como parte intrínseca de la operación de estos sistemas, son esenciales para numerosos propósitos, incluidas las perspectivas estacionales para el suministro de agua para uso doméstico, irrigación, producción de energía hidroeléctrica, navegación, protección del medio ambiente y para la mitigación de desastres naturales como inundaciones y sequías (Coulibaly 2000). Al mismo tiempo, obtener pronósticos de caudales de alta calidad y el

uso adecuado de estos pronósticos permite obtener beneficios económicos y sociales considerables (Dong *et al.* 2006; Abudu 2009). Estos beneficios incluyen la mejora de la calidad del agua y condiciones de navegación, protección de la fauna y la restauración del medio ambiente, la reducción de los gastos en la mitigación de inundaciones y la gestión de sequías, y la protección de la vida humana y la propiedad (Abudu 2009).

Por otra parte, la calidad de los pronósticos de caudales se cuantifica en términos de “tiempo de anticipación” (*lead time* en inglés) y precisión. El primero se refiere al intervalo de tiempo entre la fecha de emisión del pronóstico y la ocurrencia del evento de caudal pronosticado. La precisión del pronóstico de caudal se puede definir como la diferencia entre el caudal pronosticado y el real (Dong *et al.* 2006). En ese sentido, el pronóstico de caudales juega un papel cada vez más importante en la mitigación de inundaciones, y la gestión y planificación de los recursos hídricos. Particularmente, el pronóstico a corto plazo tales como a escalas de tiempo horario y diario es crucial para la alerta y defensa de inundaciones. Además, un mejor entendimiento del proceso hidrológico es fundamental para mejorar la habilidad del pronóstico de caudales (Wang 2006; Abudu 2009).

Debido a la importancia de los pronósticos hidrológicos, un considerable número de modelos y metodologías orientados a pronósticos de caudales (o descargas), se han estudiado, desarrollado y aplicado desde hace varias décadas. De ahí que, enfoques actuales se pueden clasificar en dos categorías generales: modelos “basados en procesos” y modelos “basados en datos” (Wang 2006; Di *et al.* 2014). Los modelos de la primera categoría consideran principalmente los mecanismos físicos internos de los procesos hidrológicos y, por lo general, necesitan una gran cantidad de datos para la calibración y validación (Lavado *et al.* 2009; Zubieta *et al.* 2015, 2017), sin embargo, no siempre hay suficientes datos disponibles.

Por el contrario, los modelos “basados en datos” son esencialmente modelos de caja negra, que identifican matemáticamente la relación entre las entradas y las salidas, sin considerar el mecanismo físico interno del sistema de la cuenca de interés (Wang 2006). Recientemente, esta categoría de modelamiento, se ha vuelto muy popular en el pronóstico de caudales, debido al aumento de la disponibilidad de los datos de las estaciones de medición, obtención de datos en tiempo real, y el aumento de la capacidad computacional con el desarrollo de métodos más robustos y técnicas informáticas (Abudu 2009). Además,

se ha demostrado que tienen la ventaja de una formulación más sencilla que los modelos “basados en procesos” (Wang 2006).

Tradicionalmente, los modelos autoregresivos de medias móviles (ARMA) propuestos por Box y Jenkins (1970), han sido ampliamente utilizados para el modelamiento de series temporales de recursos hídricos (Box *et al.* 1976; Hipel y McLeod 1994; Maier y Dandy 1996). Estos modelos asumen que la serie temporal es estacionaria, lineal y siguen una distribución estadística conocida, tal como la distribución normal. Pero, pueden ser totalmente inapropiadas si el mecanismo subyacente no es lineal, de hecho, los sistemas del mundo real son a menudo no lineales (Zhang *et al.* 1998; Zhang 2003; Wang *et al.* 2009).

Los métodos no lineales “basados en datos” tales como las redes neuronales artificiales (RNA) son ampliamente utilizadas para superar muchas dificultades en el modelamiento de series de tiempo de variables hidrológicas, incluyendo el pronóstico de caudales y se han convertido en una de las herramientas más prometedoras, ganando cada vez más popularidad en las últimas décadas. La RNA es una estructura matemática flexible que es capaz de identificar relaciones no lineales complejas entre los conjuntos de datos de entrada y salida sin la necesidad de comprender la naturaleza de los fenómenos (ASCE Task Committee 2000a, b; Wang 2006). Sin embargo, un problema con estos métodos no lineales es que tienen limitaciones con datos no estacionarios, y no son muy satisfactorios en términos de precisión porque consideran sólo unos pocos aspectos del comportamiento de la serie temporal (Wang y Ding 2003; Adamowski 2008). Por tanto, existe una necesidad de mejorar estas técnicas de pronóstico.

En la última década, el análisis de ondeleta (“*wavelet*”) se ha investigado en una serie de disciplinas y se ha encontrado que es muy efectivo con datos no estacionarios, de modo que, se ha convertido en una técnica útil para analizar las variaciones, periodicidades y tendencias en las series temporales (Adamowski 2008; Dadu y Deka 2013). El análisis *wavelet* es un análisis multiresolución en el dominio de tiempo y frecuencia y tienen ventajas sobre los métodos tradicionales de *Fourier* en el análisis de situaciones físicas donde la señal contiene discontinuidades y picos agudos, que son comunes en hidrología (Sivakumar y Berndtsson 2010; Deka y Prahlada 2012; Nourani *et al.* 2009b). Aunque las aplicaciones de *wavelets* en hidrología comenzaron en la década de 1990 y han continuado a un ritmo muy rápido (Kumar y Foufoula-Georgiou 1993; Labat 2008).

Uno de los recientes avances en el pronóstico de series de caudales se basa en el acoplamiento de modelos de la inteligencia artificial (IA) con el análisis *wavelet*, con la finalidad de aumentar la precisión de los pronósticos (Sivakumar y Berndtsson 2010). No obstante, en la mayoría de los modelos híbridos, la transformada de *wavelet* (TW) se utiliza como técnica de preprocesamiento que ayudan a mejorar el performance del modelo mediante la captura de información útil sobre varios niveles de resolución (Adamowski 2008). Asimismo, se ha encontrado que la hibridación de la TW con modelos lineales y no lineales, mejoran los resultados significativamente (Deka y Prahlada 2012; Nourani *et al.* 2014a). En suma, entre las áreas y problemas de la hidrología donde se han empleado las *wavelets* son variados, incluyendo el pronóstico de caudales, cuyos resultados son sin duda alentadores, ya que indican la utilidad de la TW para el estudio de las series hidrológicas (Sivakumar y Berndtsson 2010).

El pronóstico de caudales es desafiante debido a la gran complejidad de los sistemas hidrológicos. La mejora de la calidad de pronósticos de caudales siempre ha sido una tarea importante y motivo de gran preocupación para los investigadores y los predictores hidrológicos. Shamsheldin (2004) señala que no existe un método de pronóstico de caudales único que proporciona óptimos resultados de pronóstico en todas las circunstancias. Asimismo, ningún modelo de pronóstico individual es poderoso y lo suficientemente general como para superar a los demás para todo tipo de cuencas y en todas las circunstancias. Ya que cada uno de las técnicas de pronóstico tiene sus propias ventajas y desventajas.

Tomando en cuenta estas aseveraciones, la presente investigación tiene por objetos desarrollar modelos de pronóstico de caudales diarios a corto plazo del río Amazonas, mediante dos técnicas de la inteligencia artificial clásicos (RNA) e híbridos *wavelet* redes neuronales (WRN) con diferentes estructuras y niveles de descomposición *wavelet*. Comparando el desempeño de los modelos, se determinará el modelo de mejor ajuste para alcanzar los resultados más exactos. En suma, el alcance final de esta investigación es proporcionar herramientas más efectivas a los tomadores de decisiones en la planificación de los recursos hídricos de manera sostenible, protección contra las inundaciones, mitigación de la contaminación o concesión de licencias hídricas con fines de explotación.

1.3. ALCANCES Y LIMITACIONES

Este estudio incluye el uso de dos técnicas de pronóstico de caudales diarios, cada uno con sus propias ventajas y desventajas, ya que no existe una técnica de modelamiento que garantice un pronóstico preciso a corto plazo. Estas técnicas de la inteligencia artificial se hallaron como un enfoque poderoso para modelar procesos hidrológicos complejos. Así, se ha elegido las redes neuronales artificiales (RNA) convencionales; como un método no lineal “basados en datos”, y el enfoque híbrido *wavelet* redes neuronales (WRN); basado en una combinación de análisis *wavelet* para el preprocesamiento de datos no estacionarios y RNA. El programa de computación que se usa para los modelos de RNA e híbrido WRN es el *Matlab* (R2017a) y sus *Toolbox Wavelet* y *Neural Network*.

Para lograr los objetivos, una escala de tiempo, diario, y una variable hidrológica, caudales históricos registrados en la estación hidrométrica de Tamshiyacu, río Amazonas, fueron seleccionados para modelar y pronosticar en esta investigación. Los datos disponibles, que es la serie temporal de caudales diarios, se obtuvieron del monitoreo desarrollado por el Servicio Nacional de Meteorología e Hidrología del Perú, como parte del observatorio ORE-HYBAM, www.ore-hybam.org.

Se realizaron las comparaciones de desempeño de pronósticos de los modelos formulados de RNA tradicionales y la capacidad potencial del enfoque de modelamiento híbrido WRN, evaluándose mediante cuatro índices estadísticos (RMSE: raíz del error cuadrático medio, MAE: error absoluto medio, MARE: error absoluto relativo medio, y NSE: eficiencia de *Nash-Sutcliffe*) y análisis gráfico de errores residuales.

Con base en el análisis estadístico de desempeño de los modelos formulados para múltiples horizontes de pronóstico, se determinaron los modelos finales más eficientes que proveen información precisa para la toma de decisiones. A fin de sugerir que podrían ser utilizados en los sistemas de pronóstico hidrológico en la cuenca del río Amazonas.

Para desarrollar modelos de pronósticos más complejos y precisos, así mejorar la capacidad predictiva de los modelos de RNA tradicionales, las siguientes consideraciones y procedimientos se utilizaron:

- Para construir los diversos modelos, previamente, se analizó la predictibilidad del comportamiento del caudal diario, a través de la correlación entre el valor presente y los valores pasados. Los retardos considerados para las entradas a los modelos son en escala de tiempo diario, hasta un máximo de seis días antecedentes.

- Con el fin de minimizar el error de pronóstico estimado, todos los valores predictores utilizados en el modelamiento fueron valores observados antecedentes; no fueron utilizados valores pronosticados en ninguno de los procedimientos de modelamiento.
- Modelos de RNA convencionales usando la arquitectura de *perceptrón* múltiples capas *feedforward* fueron utilizados para el pronóstico de caudales diarios a corto plazo, que a la vez sirvieron como modelos de referencia (*Benchmark* en inglés) para la comparación con los modelos híbridos WRN formulados en esta investigación.
- Los modelos híbridos WRN tratan esencialmente de una combinación de dos técnicas, la transformada de *wavelet* discreta y RNA. Para el buen desempeño de estos modelos se consideraron varios factores, como la selección de la función *wavelet* madre *Daubechies* (db2), cuarto nivel de descomposición, uso de todas las subseries descompuestas y la selección de RNA de *perceptrón* multicapa.
- También, se compararon los resultados de pronóstico hidrológico para múltiples horizontes de pronóstico (sintetizado hasta los 30 días de anticipación), para evaluar la capacidad predictiva de los modelos formulados, a través de cuatro parámetros estadísticos: RMSE, MAE, MARE y NSE.

Durante el proceso de investigación, se han identificado factores limitantes, relacionado a los siguientes aspectos:

- El estudio actual está limitado a una sola variable hidrológica, en consecuencia, se formularon modelos univaridos. La influencia de caudales registrados en estaciones hidrométricas aguas arriba y la relación con otras variables hidroclimáticas no se tomaron en cuenta en este estudio.
- La simulación y pronóstico de caudales está restringido a un solo sitio. Los datos disponibles, en la estación hidrométrica Tamshiyacu en el río Amazonas, se limitan a los años 1984 y 2013, que se utilizan para la aplicación de los métodos de pronóstico hidrológico en esta investigación.
- El pronóstico a corto plazo basado sobre escala temporal diario es de interés en este estudio para ser simulado y pronosticado. Sin embargo, la aplicabilidad y el desempeño de los modelos propuestos a escalas temporales más grandes no se discuten aquí.

- El análisis de incertidumbre de los modelos, como un componente clave para determinar la fiabilidad de los pronósticos y crítica al momento de utilizar los resultados en la toma de decisiones, tampoco fue cubierto en el presente estudio.

1.4. OBJETIVOS

1.4.1. Objetivo general

Desarrollar modelos “basados en datos” de la inteligencia artificial con un enfoque híbrido (*wavelet* y redes neuronales) para simular y pronosticar caudales diarios del río Amazonas (en la estación de Tamshiyacu) a múltiples días de anticipación.

1.4.2. Objetivos específicos

- Desarrollar, simular y validar modelos univariados de redes neuronales artificiales (RNA) estándares para el pronóstico hidrológico a un día posterior, utilizando como entradas a la red sólo rezagos de la misma serie de caudales diarios.
- Encontrar la estructura de la RNA más robusta para el pronóstico a múltiples días de anticipación, evaluando sus desempeños mediante índices estadísticos y análisis gráfico de errores residuales. Además, establecerlo como el modelo de referencia (*Benchmark*) para la comparación con los modelos híbridos.
- Desarrollar y evaluar la efectividad de los modelos híbridos *wavelet* redes neuronales (WRN) en la precisión de pronósticos a múltiples días de anticipación, explorando el impacto de los niveles de descomposición y la *wavelet* madre *Daubechies* de orden superior; y
- Comparar el desempeño de las dos técnicas aplicadas, las RNA y el enfoque híbrido en el pronóstico de caudales diarios a múltiples días de anticipación (corto plazo), a través de los índices estadísticos RMSE (raíz del error cuadrático medio) y NSE (eficiencia de *Nash-Sutcliffe*).

II. REVISIÓN DE LITERATURA

A lo largo de este Capítulo se presenta el conjunto de teorías y metodologías en las que se fundamentan las distintas técnicas que se han utilizado en la simulación y pronóstico de series de tiempo hidrológicas. Se intenta presentar la información con un sentido más profundo al que se abordará en las secciones posteriores. Este conjunto de teorías acota el área de trabajo de nuestra investigación posterior, por lo que se tratará de definir un conjunto de conceptos y proposiciones que permitan abordar el problema desde una base metodológica sólida.

2.1. CONCEPTOS BÁSICOS

2.1.1. Descripción y análisis preliminar

Consiste en describir las principales características de la serie utilizando estadísticos resumen y/o métodos gráficos. Cuando se obtiene una serie temporal, el primer paso en el análisis es usualmente graficar las observaciones a lo largo del tiempo para obtener medidas descriptivas simples de las principales características de la serie. Este gráfico mostrará rasgos o propiedades de la serie tales como tendencia, estacionalidad, *outliers*, discontinuidades y cambios graduales o bruscos.

Por todo lo anterior, el ploteo de los datos es vital para describir los datos y como ayuda en la formulación de un modelo sensible. Además, este grafico puede ayudar a decidir si los datos necesitan ser transformados antes del análisis. Por ejemplo, si se observa una tendencia ascendente en el gráfico de los datos y la varianza aumenta conforme lo hace el valor medio.

Entonces, una transformación puede ser recomendable para estabilizar la varianza. La transformación de los datos también es recomendable en el caso de que las observaciones presenten algún sesgo, asimetría y/o cuando los cambios en la varianza son severos. En estos casos es recomendable una normalización de los datos. Finalmente, si los efectos estacionales se observan multiplicativos, entonces sería deseable transformar los datos para convertir en aditivo el efecto estacional.

2.1.2. Definición de modelamiento y simulación

Los problemas de toma de decisiones, tanto en el diseño y operación de sistemas de recursos hídricos a gran escala, como en los embalses de control de inundaciones, canales, acueductos, sistemas de suministro de agua y otros han dado lugar a una necesidad de enfoques matemáticos tales como la simulación y síntesis para investigar el proyecto total.

a. Modelamiento

El modelamiento es el proceso de producir un modelo; un modelo es una representación de la construcción y funcionamiento de algún sistema de interés. Un modelo es similar pero más simple que el sistema que representa. Un propósito de un modelo es permitir al analista predecir el efecto de cambios en el sistema. Por una parte, un modelo debe ser una aproximación cercana al sistema real e incorporar la mayoría de sus características más sobresalientes. Por otra parte, no debe ser tan complejo que es imposible comprender y experimentar con ello. Un buen modelo es una compensación razonable entre el realismo y la sencillez. Practicantes de simulación recomiendan aumentar la complejidad de un modelo de manera iterativa. Una cuestión importante de modelamiento es la validez del modelo. Técnicas de validación de modelos incluyen la simulación del modelo en condiciones de entrada conocida y comparando los resultados del modelo con la salida del sistema.

Generalmente, un modelo destinado a un estudio de simulación es un modelo matemático desarrollado con la ayuda de *software* de simulación. Clasificaciones de modelo matemático incluyen determinístico (variables de entrada y salida son valores fijos) o estocástico (al menos una de las variables de entrada o de salida es probabilística); estático (el tiempo no se tiene en cuenta) o dinámico (interacciones variables en el tiempo entre las variables se tienen en cuenta). Típicamente, los modelos de simulación son estocásticos y dinámicos (Maria 1997).

b. Simulación

La simulación se define como la descripción matemática de la respuesta de un sistema hidrológico de recursos hídricos a una serie de eventos durante un período de tiempo seleccionado. Por ejemplo, la simulación puede significar el cálculo diario, mensual o estacional del caudal basado en la precipitación; o calcular el hidrograma de descarga

resultante de una tormenta conocida o hipotética; o simplemente completar los valores que faltan en un registro de caudales (Viessman *et al.* 1989).

Una simulación de un sistema es el funcionamiento de un modelo del sistema. El modelo puede ser reconfigurado y experimentado; usualmente, esto es imposible, demasiado caro o poco práctico hacer en el sistema que representa. El funcionamiento del modelo puede ser estudiado, y, por lo tanto, las propiedades referentes al comportamiento del sistema real o su subsistema pueden deducirse. En su sentido más amplio, la simulación es una herramienta para evaluar el desempeño de un sistema, existente o propuesta, bajo diferentes configuraciones de interés y durante largos períodos de tiempo real (Maria 1997).

Además, la simulación se utiliza antes de que se altere un sistema existente o se construya un nuevo sistema, para reducir las probabilidades de incumplimiento de las especificaciones, para eliminar los cuellos de botella imprevistos, para evitar bajo o sobreutilización de los recursos, y para optimizar el performance del sistema.

2.1.3. Pronóstico hidrológico

El pronóstico hidrológico es la estimación previa de los estados futuros de los fenómenos hidrológicos en tiempo real, que no debe confundirse con el cálculo de datos de diseño hidrológico (predicción). Son esenciales para la operación eficiente de la infraestructura hídrica y la mitigación de los desastres naturales tales como inundaciones y sequías. Además, se están convirtiendo cada vez más importante en el apoyo a la gestión integrada de los recursos hídricos y la reducción de pérdidas por inundación inducida (Nemec 1986; WMO168 2009).

Describir y predecir estados futuros del agua se puede categorizar sobre la base de cuán lejos en el futuro se prevé que ocurra el evento. Por ejemplo, las previsiones para diversos elementos hidrológicos tales como descargas, niveles y velocidades pueden hacerse desde el inicio del pronóstico hasta diferentes momentos en el futuro (WMO168 2009).

a. Métodos de pronóstico

A pesar de la amplia gama de situaciones problemáticas donde se requieren pronósticos con el fin de tomar buenas decisiones, sólo hay dos grandes tipos básicos de técnicas de pronóstico: métodos cualitativos y métodos cuantitativos (Figura 2). Los métodos de

pronóstico cualitativos son aquellos que utilizan la intuición y la experiencia para predecir eventos futuros subjetivamente. Los métodos de pronóstico cuantitativos implican el análisis de datos históricos en un intento por identificar patrones que pueden ser usados para describirlos.

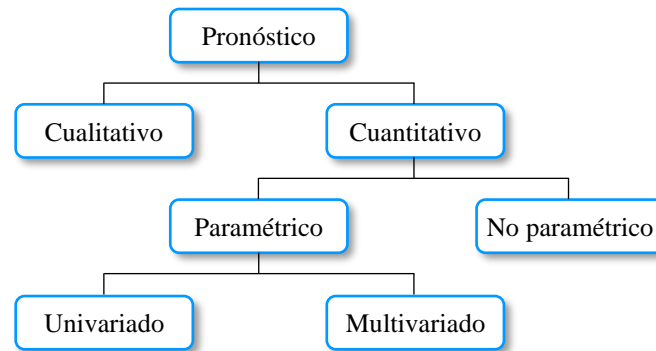


Figura 2: Tipos básicos de métodos de pronóstico

FUENTE: Adaptado de [Blake et al. \(1995\)](#)

Estas descripciones o modelos pueden ser utilizados entonces para extrapolar el comportamiento de los datos en el futuro. Si el modelo contiene constantes desconocidas a ser estimado durante el proceso de pronóstico, se dice que este modelo es un modelo paramétrico, siendo los constantes estimados sus parámetros. Un modelo paramétrico es particularmente atractivo si los parámetros representan características en el mundo físico. Si el modelo involucra más de una variable predictiva relacionada con la variable de respuesta, se llama un modelo multivariado ([Blake et al. 1995](#); [Montgomery et al. 2015](#)).

Los pronósticos de una variable se basan en un modelo ajustado solamente a observaciones pasadas y presentes de una serie temporal, posiblemente aumentado por una función simple en el tiempo, tal como una tendencia lineal global. Esto significa, por ejemplo, que los pronósticos univariados del nivel de una presa se basarán enteramente en los niveles pasados de la misma y no tomarían en cuenta otros factores hidrológicos o climáticos como podrían ser el caudal en el río que la abastece o la precipitación ([Espinoza 2010](#)).

En muchas ocasiones las observaciones se registran simultáneamente para varias series temporales. Por citar un ejemplo, en meteorología se registran la temperatura, presión barométrica y lluvia en el mismo sitio para la misma secuencia de puntos en el tiempo. Dados estos datos se podría intentar desarrollar un modelo multivariado para describir las interrelaciones entre las series, y usar este modelo para hacer pronósticos. Con datos de series temporales, el proceso de modelación se complica por la necesidad no solo de

modelar la interdependencia entre las series, sino también la dependencia dentro de cada serie (Espinoza 2010).

Ajustar un modelo multivariado a una serie temporal no es fácil, aun con los grandes avances en recursos computacionales que se han conseguido en los últimos años. Los modelos univariados pueden ser útiles para muchos fines y es obvio que los modelos multivariados ofrecen mucho en la parte de la comprensión de la estructura subyacente de un sistema dado y también en ocasiones en la obtención de mejores pronósticos. Sin embargo, no todo el tiempo esta afirmación es cierta. Los modelos multivariados normalmente consiguen un mejor ajuste que los modelos univariados, pero hay varias razones por las cuales la obtención de mejores ajustes no necesariamente se refleja, aunque en ocasiones lo hacen en mejores pronósticos.

b. Categorías de pronóstico hidrológico

Enfoques actuales de pronóstico hidrológico se pueden dividir en dos categorías: los modelos “basados en procesos” y los modelos “basados en datos” (Figura 3). Los modelos de la primera categoría consideran principalmente los mecanismos físicos internos de los procesos hidrológicos, y por lo general necesitan una gran cantidad de datos para la calibración y validación. Sin embargo, no siempre hay suficientes datos disponibles. Los modelos “basados en datos” se conocen como métodos de caja negra, y no tienen en cuenta el proceso hidrológico física, en su lugar identifican la relación entre las entradas y las salidas matemáticamente. Los modelos “basados en datos” han demostrado tener la ventaja de las demandas más bajas para los datos cuantitativos, mejor desempeño de predicción y formulación más simple que los modelos “basados en procesos” (Di *et al.* 2014).

Los modelos “basados en datos” (*data-driven* en inglés) desarrollados en las últimas décadas contienen dos categorías principales: las técnicas estadísticas tradicionales y herramientas de la inteligencia artificiales - IA (Figura 3). Los modelos estadísticos pueden proporcionar buenos resultados de pronóstico cuando la serie son lineal o casi lineal, pero no pueden captar los patrones no lineales ocultos en series de tiempo hidrológicas. Los modelos no lineales y la IA incluyen redes neuronales artificiales (RNA), algoritmos genéticos (AG) y máquinas de vectores soporte (MVS), que proporcionan soluciones poderosas para el pronóstico hidrológico no lineales. Sin embargo, estos métodos de la IA tienen sus propios defectos y desventajas. Por ej., RNA menudo sufre de sobreajuste y MVS suelen ser sensibles a la selección de parámetros (Di *et al.* 2014).

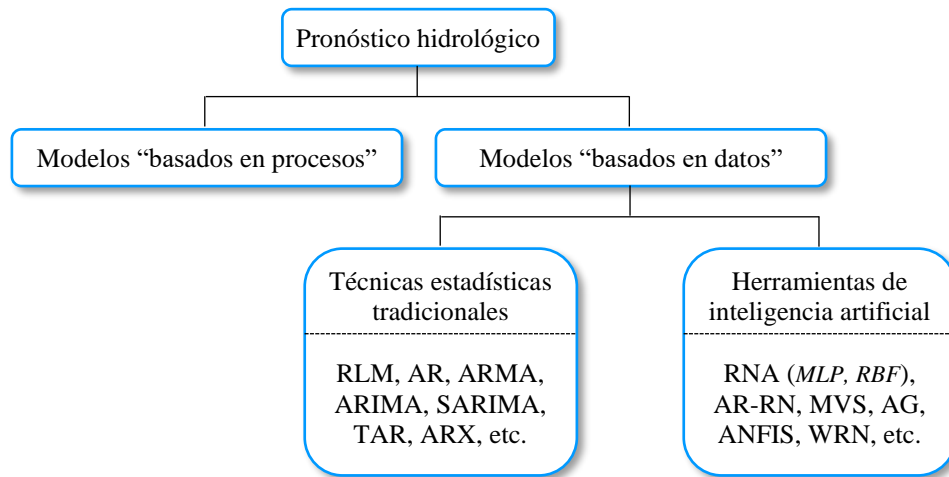


Figura 3: Principales categorías de pronóstico hidrológico
FUENTE: Esquemático de *Di et al. (2014)*

c. Horizontes de pronóstico

Los problemas de pronóstico a menudo son clasificados como corto plazo, mediano plazo y largo plazo (Figura 4). Los problemas de pronóstico a corto plazo implican predecir eventos sólo unos pocos períodos de tiempo (días, semanas y meses) en el futuro. Los pronósticos a mediano plazo se extienden de uno a dos años en el futuro, y los problemas de pronóstico a largo plazo pueden extenderse más allá de eso por muchos años. Los pronósticos a corto y mediano plazo se utilizan para la gestión de operaciones y desarrollo de proyectos, mientras que los pronósticos a largo plazo se pueden utilizar para la planificación estratégica. El pronóstico a corto y mediano plazo se basa normalmente en la identificación, modelamiento y la extrapolación de los patrones encontrados en los datos históricos. Debido a que estos datos históricos por lo general presentan una inercia y no cambian drásticamente muy rápidamente, los métodos estadísticos son muy útiles para el pronóstico a corto y medio plazo (*Montgomery et al. 2015*).

Si bien el pronóstico a corto plazo tales como el pronóstico horario o diaria es crucial para la alerta y defensa de inundaciones, el pronóstico a largo plazo basada sobre escalas temporales mensuales, estacionales o anuales es muy útil en operaciones de embalses y decisiones de gestión de riego, tales como comunicados de programación, asignación de agua para usuarios río abajo, mitigación de sequías y la gestión de tratados de ríos o ejecución de cumplimiento de pactos (*Abudu 2009*).

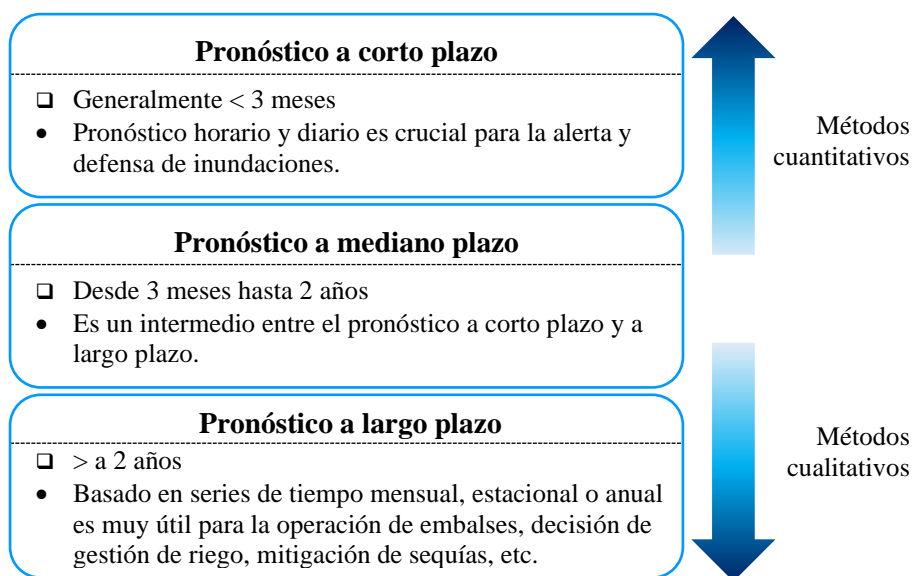


Figura 4: Horizontes de pronóstico y métodos básicos de pronóstico
FUENTE: Esquematizado de [Abudu \(2009\)](#) y [Montgomery et al. \(2015\)](#)

Es obvio que uno de los criterios más importantes de la clasificación es el “tiempo de anticipación” (*lead time* en inglés). Su definición no es universalmente aceptada, pero un gran consenso puede suponer en torno a la siguiente: El tiempo de anticipación es el tiempo entre la emisión del pronóstico de un fenómeno hidrológico y su ocurrencia. Este tiempo de anticipación puede ser considerado como el tiempo de anticipación neto, un tiempo de anticipación bruto sería el tiempo entre la ocurrencia de la causa(s) de un fenómeno hidrológico y su ocurrencia (es decir, entre la ocurrencia de una lluvia que causa una inundación y la ocurrencia del pico de la crecida). Sin embargo, no se recomienda el uso del tiempo de anticipación bruto, ya que es bastante grande la incertidumbre acerca de los tiempos de ocurrencias de los fenómenos causales. La relatividad de la ocurrencia temporal con ocurrencia espacial ha de ser también tomado en cuenta, como en el pronóstico de inundaciones, en particular, un plazo de anticipación relativamente corto para un tramo de río corto puede convertirse en un plazo de anticipación largo para los puntos muchos más aguas abajo ([Nemec 1986](#)). La Figura 5 ilustra los conceptos anteriores.

Idealmente, la elección de un tiempo de anticipación de pronóstico depende de las finalidades del pronóstico a ser utilizado. Pronósticos hidrológicos a corto plazo, por ejemplo, pueden ser utilizados para reducir el potencial daño por inundación mientras que los pronósticos a más largo plazo se utilizan para optimizar la gestión de los recursos hídricos ([Hingray et al. 2015](#)).

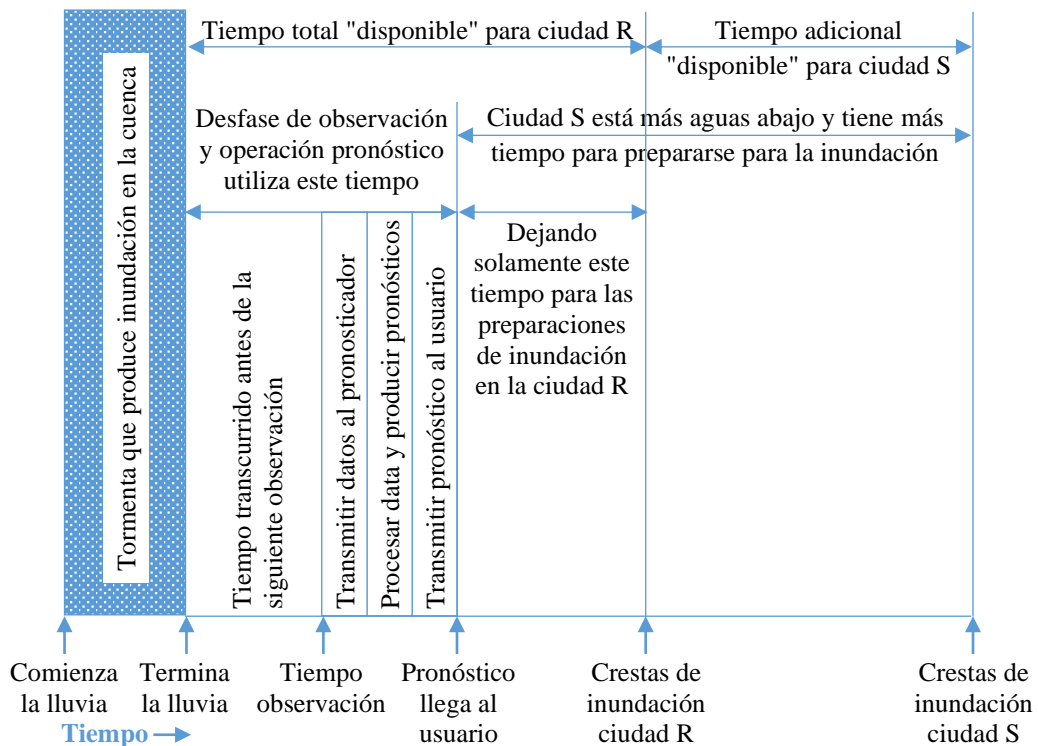


Figura 5: Componentes de tiempo y espacio del "tiempo de anticipación"
FUENTE: Adaptado de Nemeč (1986)

2.1.4. Control de un proceso

Si se consigue construir un modelo que entregue buenos pronósticos, se puede permitir al analista pasar a controlar un proceso dado. Es decir, los pronósticos aportan elementos para tomar decisiones de control óptimo de un proceso. Por ejemplo, para el caso de una serie temporal que mide las presiones registradas en una red de abastecimiento de un sector, el propósito del análisis podría ser mantener las presiones operando dentro de un rango adecuado para las tuberías existentes en el sector. Los problemas de control están muy relacionados con el pronóstico en muchas situaciones. Por ejemplo, si se predice que las presiones del sector van a alejarse del objetivo, entonces se deben tomar medidas correctivas (Espinoza 2010).

2.2. REDES NEURONALES ARTIFICIALES

Las redes neuronales artificiales (en adelante RNA) quedan englobadas conjuntamente con la lógica difusa y los sistemas expertos dentro de las denominadas técnicas de la inteligencia artificial (IA). En los últimos años, la inteligencia artificial está encontrando aplicaciones en numerosas actividades cotidianas de los seres humanos y además ha encontrado un importante campo de aplicación para desarrollarse en áreas del

conocimiento científico, tales como la robótica, la visión artificial, técnicas de aprendizaje, gestión del conocimiento, modelación de procesos físicos, etc.

2.2.1. Definición

Las RNA, o simplemente redes neuronales, surgen como un intento para modelar la actividad cerebral. Hoy en día las mismas son tratadas de forma más abstracta: redes que interconectan, de forma no lineal, elementos de cálculo. Son modelos paramétricos que pueden ser utilizados para aproximar cualquier tipo de relación funcional entre variables predictoras y de respuesta. A diferencia de lo que sucede con, por ejemplo, los modelos lineales no es necesario predefinir el tipo de relación entre las variables a relacionar. Los datos observados se utilizan para entrenar la red, la cual aprende a aproximar la relación entre predictores y respuestas adaptadas, de forma iterativa, sus parámetros. De entre los problemas en los que suelen aplicarse RNA destacan el reconocimiento de patrones y problemas de predicción, especialmente casos con alta dimensionalidad y grandes cantidades de datos ([Talento 2011](#)).

Es difícil encontrar una definición unificada para las RNA. Una que cubre muchos de los aspectos mencionados por la mayoría de los autores podría ser como la expone [Espinoza \(2010\)](#), que la define como un procesador distribuido en paralelo compuesto por unidades de proceso elementales (nodos o neuronas artificiales) masivamente interconectadas entre sí y con organización jerárquica, con capacidad de adquirir conocimiento de tipo experimental con capacidad para simular y predecir procesos no lineales.

2.2.2. Breve reseña histórica de las redes neuronales

El concepto y la idea original de RNA aparecen por primera vez en una publicación pionera presentada por *McCulloch* y *Pitts* en 1943. Surge con la idea de emular y comprender las funciones que realiza el cerebro humano y su sistema nervioso por medio de una conceptualización del mismo en un modelo matemático de estructuras simples (neuronas) interconectadas ([Espinoza 2010](#)). A partir de esta primera publicación fueron apareciendo varias más. Tras una evolución de las RNA en 1951, *Minsky* y *Edmonds* construyen la primera máquina de redes neuronales y en 1960, *Widrow* y *Hoff* son los primeros en utilizar una RNA en la aplicación en un caso real ([Pujol 2009](#)).

Sin embargo, en 1969 se produce un bloqueo en la utilización de las RNA, aunque es hasta la década de los ochenta cuando el tema de las RNA tiene un reimpulso con la publicación de [Rumelhart et al. \(1986\)](#). En ella se introduce el algoritmo de retropropagación del error que es el más utilizado hasta la actualidad con grandes evoluciones ya que representa un marco teórico matemático riguroso para las RNA. Un buen resumen de la evolución histórica de las RNA se encuentra en [Haykin \(1998\)](#), así como en [Hilera y Martínez \(1995\)](#).

La utilización de las RNA en ingeniería civil viene de los años 80, pero sólo a partir de 1991, aparecen las primeras aplicaciones en hidrología, como predicción de lluvia, consumo de agua y estimación de caudales y predicción de caudales. Para los hidrólogos es una técnica joven, pero ha tenido una rápida aceptación y una expansión positiva en publicaciones y conferencias ([Pujol 2009](#)).

2.2.3. La neurona biológica

La célula nerviosa o neurona biológica es una de las estructuras fundamentales del sistema nervioso central. Está constituida, generalmente, por dendritas o terminales nerviosos, el cuerpo de la célula y el axón como se detalla en la Figura 6. El axón en muchas neuronas está rodeado por una membrana que se denomina vaina de mielina la cual funciona como un conductor eléctrico encargado de conducir los potenciales eléctricos entre neuronas.

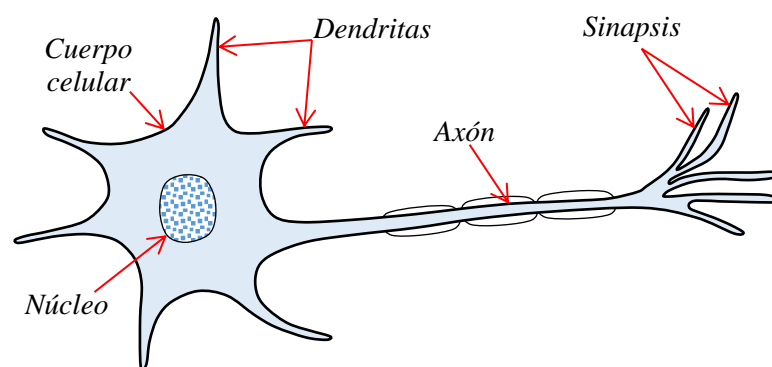


Figura 6: Modelo simplificado de una neurona biológica
FUENTE: Adaptado de [Salinas \(2009\)](#)

La sinapsis es el proceso en el cual se conecta el axón de una neurona con las distintas partes de otras neuronas, esto se logra usando las propiedades iónicas de la membrana celular, la cual actúa manteniendo una diferencia de potencial entre el fluido intracelular y el fluido extracelular.

Este efecto se consigue mediante la acción de la denominada bomba de sodio potasio, que está determinada por las concentraciones altas de potasio dentro de la célula y concentraciones altas de sodio fuera de ella, este equilibrio entre el sodio y potasio produce una diferencia de potencial denominado potencial de reposo de la célula. Cuando se produce excitación entre una o más neuronas, una cantidad de iones de sodio penetran en la célula produciendo despolarización generando el denominado potencial de acción que es un impulso eléctrico que dispara las neuronas (Salinas 2009).

2.2.4. La neurona artificial

La neurona artificial fue diseñada para "emular" las características del funcionamiento básico de la neurona biológica. En esencia, se aplica un conjunto de entradas a la neurona, cada una de las cuales representa una salida de otra neurona. Cada entrada se multiplica por su "peso" o ponderación correspondiente análoga al grado de conexión de la sinapsis. Todas las entradas ponderadas se suman y se determina el nivel de excitación o activación de la neurona (Basogain 1998). Una representación vectorial del funcionamiento básico de una neurona artificial se indica según la siguiente expresión de la ecuación 2.1.

$$Z_j = \sum w_{ij}x_j - \theta_i \quad (2.1)$$

Siendo Z_j la salida, x el vector de entrada, w el vector de pesos y θ_i el umbral.

Normalmente la señal de salida Z_j suele ser procesada por una función de activación $f()$ para producir la señal de salida de la neurona y_i . La función $f()$ puede ser una función lineal, o una función umbral o una función no lineal que simula con mayor precisión las características de transferencia no lineales de las neuronas biológicas. La Figura 7 representa una neurona artificial con una función de activación $f()$.

Este tipo de modelo de neurona artificial ignora muchas de las características de las neuronas biológicas. Entre ellas destaca la omisión de retardos y de sincronismo en la generación de la salida. No obstante, a pesar de estas limitaciones las redes construidas con este tipo de neurona artificial presentan cualidades y atributos con cierta similitud a la de los sistemas biológicos (Basogain 1998).

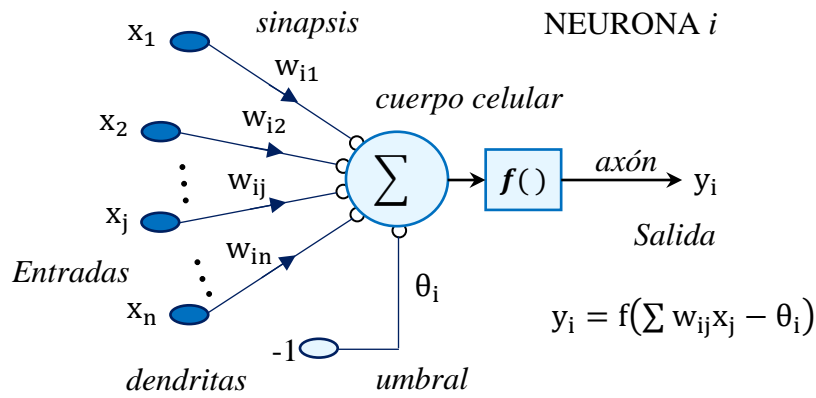


Figura 7: Modelo de una neurona artificial
FUENTE: Adaptado de [Basogain \(1998\)](#)

2.2.5. Elementos básicos de una red neuronal

En la Figura 8, puede apreciarse un esquema de una red neuronal artificial de tres capas con conexiones hacia delante:

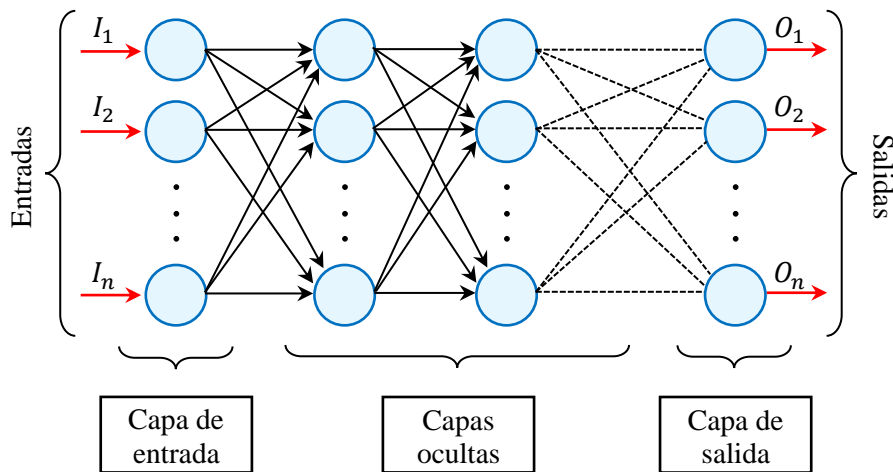


Figura 8: Elementos de una red neuronal artificial
FUENTE: Adaptado de [Socha y Ortiz \(2005\)](#)

La misma está constituida por conjunto de conexiones o pesos sináptico entre neuronas (interconexiones) y arregladas en tres capas (este último puede variar). Los datos ingresan por medio de la “capa de entrada”, pasan a través de la “capa oculta” y salen por la “capa de salida”. Cabe mencionar que la capa oculta puede estar constituida por varias capas.

2.2.6. Funciones de activación más populares

La función de activación también llamado función de transferencia. Ello determina la relación entre las entradas y salidas de un nodo y una red. En general, la función de activación introduce un grado de no linealidad que es valiosa para la mayoría de aplicaciones de RNA. En términos generales, cualquier función diferenciable puede calificar como una función de activación en la teoría. En la práctica, sólo un pequeño número de "buen comportamiento" (acotada, monótonamente creciente, y diferenciables) funciones de activación son usados (Zhang *et al.* 1998).

A continuación, se describen las funciones de activación más populares:

- **Funciones lineales:** Son aquellos que dan una salida lineal:

$$f(x) = x \quad (2.2)$$

$$f(x) = \begin{cases} -1 & \text{si } x \leq -1 \\ x & \text{si } -1 < x < 1 \\ 1 & \text{si } x \geq 1 \end{cases} \quad (2.3)$$

- **Funciones escalón:** Son aquellos que dan una salida binaria dependiente de si el valor de entrada está por encima o por debajo del valor umbral.

$$f(x) = \text{sgn}(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases} \quad (2.4)$$

- **Funciones sigmoidales:** Son aquellas monótonas acotadas que dan una salida gradual no lineal para entradas. Las funciones sigmoidales más populares son:

- La función logística [0 a 1]:

$$f(x) = \frac{1}{(1 + e^{-x})} \quad (2.5)$$

- La función tangente hiperbólica de [-1 a 1]:

$$f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \quad (2.6)$$

- **Función seno o coseno:**

$$f(x) = \text{sen}(x) \text{ ó } f(x) = \text{cos}(x) \quad (2.7)$$

Aunque existen otras funciones de activación, se suelen utilizar la sigmoide y la tangente hiperbólica. Los motivos son que, no sólo se consiguen resultados bastantes aceptables, sino que además son funciones computacionalmente poco costosas en comparación con otras existentes. Esto es debido, principalmente, a que su derivada es función de ellas mismas (Alegre 2003; Socha y Ortiz 2005).

2.2.7. Clasificación de las RNA

Respecto a la cuestión de cuántos tipos de RNA existen actualmente, se puede decir que se trata de un número inabarcable. Sin embargo, también se puede decir que del total hay aproximadamente 40 modelos que son bien conocidos por la comunidad de investigadores en RNA, más detalle en [Montaño \(2002\)](#).

a. Conexiones entre neuronas (topología)

La topología o arquitectura de una red consiste en la organización y disposición de las neuronas en la red formando capas de neuronas. Por lo tanto, una primera clasificación de las RNA que se suele hacer es en función del patrón de conexiones que presenta. Así se definen dos tipos básicos de redes:

- **Conexiones hacia adelante (*feedforward*)**

Las redes alimentadas hacia adelante generalmente conocidas como redes *feedforward* son aquellas en las que, como su nombre indica, la información se mueve en un único sentido, desde la entrada hacia la salida (Figura 9). Estas redes están clásicamente organizadas en “capas”. Cada capa agrupa a un conjunto de neuronas que reciben sinapsis de las neuronas de la capa anterior y emiten salidas hacia las neuronas de la capa siguiente. Entre las neuronas de una misma capa no hay sinapsis ([Gestal 2013](#)).

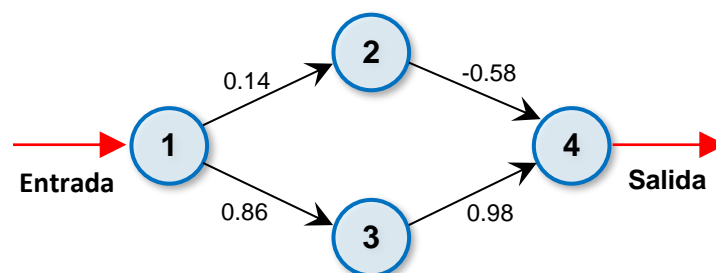


Figura 9: RNA alimentada hacia adelante
FUENTE: Adaptado de [Gestal \(2013\)](#)

En este tipo de redes existe al menos una capa de entrada, formada por las neuronas que reciben las señales de entrada a la red y una capa de salida, formada por una o más neuronas que emiten la respuesta de la red al exterior. Entre la capa de entrada

y la de salida existen una o más capas intermedias llamadas capas ocultas, pudiendo distinguir por lo tanto entre dos tipos de redes *feedforward*:

- **Monocapa:** una sola capa, ej. de este tipo de redes son el *perceptrón* y la *Adaline*.
- **Multicapa:** varias capas de neuronas, un ej. es el *perceptrón* multicapa.

En redes así construidas es evidente que la información sólo puede moverse en un sentido: desde la capa de entrada hasta la de salida, atravesando todas y cada una de las capas intermedias una sola vez. El hecho de que no haya conexión entre las neuronas de una misma capa hace que no haya tiempos de espera en los que las neuronas estén interactuando unas sobre las otras hasta que toda la capa adquiera un estado estable. Se trata por tanto de redes rápidas en sus cálculos (Gestal 2013).

- **Conexiones hacia adelante y hacia atrás (*feedforward/feedback*)**

En este tipo de redes los elementos pueden enviar estímulos a neuronas de capas anteriores, de su propia capa o a ellos mismos, por lo que desaparece el concepto de agrupamiento de las neuronas en capas (Figura 10). Cada neurona puede estar conectada a todas las demás; de este modo, cuando se recibe información de entrada a la red, cada neurona tendrá que calcular y recalcular su estado varias veces hasta que todas las neuronas de la red alcancen un estado estable. Un estado estable es aquel en el que no ocurren cambios en la salida de ninguna neurona. No habiendo cambios en las salidas, las entradas de todas las neuronas serán también constantes, por lo que no tendrán que modificar su estado de activación ni en su respuesta, manteniéndose así un estado global estable. Las redes retroalimentadas emulan más fielmente la estructura del cerebro humano, en donde los fenómenos de retroalimentación son fundamentales (Gestal 2013; Sánchez 2014).

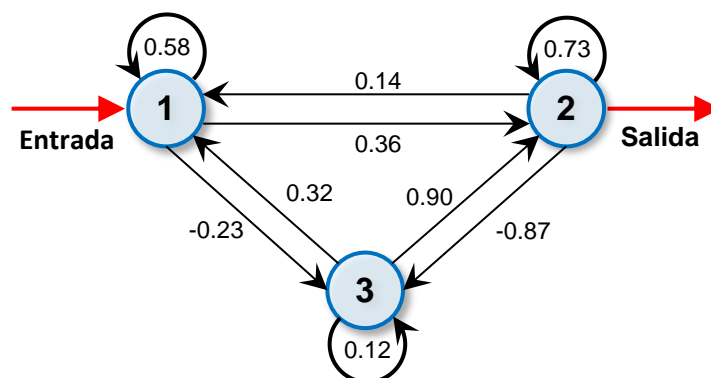


Figura 10: RNA con retroalimentación

FUENTE: Adaptado de Gestal (2013)

b. Mecanismos de aprendizaje

En general, el proceso de aprendizaje se alcanza de dos maneras. En ocasiones, las redes neuronales utilizarán información previamente establecida; y basándose en dicha información se lleva a cabo la actualización de los pesos. En otras ocasiones las redes neuronales deben aprender a establecer los pesos sin esta información. A estos tipos de aprendizaje se les denomina supervisado y no supervisado o auto organizado, respectivamente. En general, cualquier técnica de clasificación y predicción puede encontrarse en uno de estos dos grupos (Morales 2010).

- **Aprendizaje supervisado**

En el aprendizaje supervisado básicamente a la red se le presenta un conjunto de ejemplos denominado muestra de aprendizaje, junto con la salida deseada u objetivo, e iterativamente ajusta los pesos hasta que la salida de la red tienda a ser la deseada, ver la Figura 11.

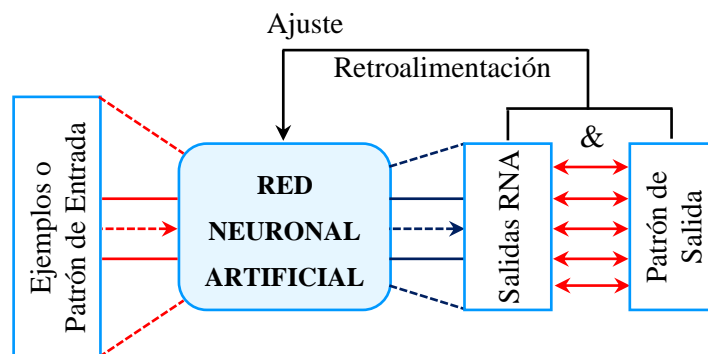


Figura 11: Aprendizaje supervisado

FUENTE: Adaptado de Morales (2010)

Con esta técnica de aprendizaje el entrenamiento consiste en presentarle a la red repetitivamente patrones de estímulos de entrada pertenecientes a un juego de ensayo. El juego de ensayo está formado por parejas “patrón de estímulos - respuesta correcta” y debe de ser elegido cuidadosamente. Cada pareja se denomina hecho. En el juego de ensayo debe estar representada equilibradamente toda la información que la red necesite aprender.

Al realizar el entrenamiento la respuesta que da la red a cada patrón se compara con la respuesta correcta ante dicho patrón y, en virtud de esa comparación, se reajustan los

pesos sinápticos. El reajuste de los pesos sinápticos está orientado a que, ante el patrón de entrada, la red se acerque cada vez más a la respuesta correcta.

Cuando ante un patrón de entrada la red de neuronas ya responde correctamente, se pasa al siguiente patrón del juego de ensayo y se procede de la misma manera. Una vez que se termina con el último patrón del juego de ensayo, se tiene que volver a empezar con el primero, ya que los pesos se han seguido modificando.

En casos sencillos, al cabo de unos pocos pasos de entrenamiento completos, con todos los elementos del juego de ensayo, los pesos sinápticos de todas las neuronas se estabilizan en torno a unos valores óptimos. Se dice entonces que el algoritmo de aprendizaje converge. Es decir, después de sucesivas presentaciones de todos los patrones estimuladores del juego de ensayo, la red, responderá correctamente a todos ellos y se puede considerar entrenada y dar por terminada la fase de aprendizaje (Gestal 2013).

- **Aprendizaje no supervisado o autoorganizado**

Por otro lado, en el aprendizaje no supervisado o autoorganizado (Figura 12), las redes son capaces de encontrar y descubrir, de manera automática, patrones de similitud dentro del conjunto de datos de entrenamiento, esto lo realiza sin ningún tipo de retroalimentación con el medio externo y sin la utilización de información a priori, cuando se utiliza este tipo de aprendizaje la red trata de determinar características de los datos del conjunto de entrenamiento: rasgos significativos, regularidades o redundancias (Morales 2010).

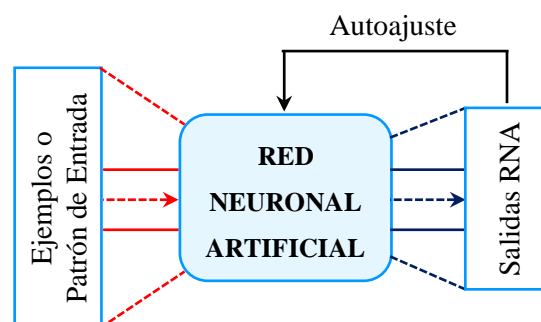


Figura 12: Aprendizaje autoorganizado
FUENTE: Adaptado de Morales (2010)

En este tipo de aprendizaje, no se especifica a la red cuál debe ser la respuesta correcta; es decir, no hay una comparación entre la respuesta de la red y la respuesta deseada. Además, en este modelo de aprendizaje no existe ninguna influencia externa a la red, puesto que no se le informa de si un resultado fue correcto o no; tan sólo se le suministran grandes cantidades de datos con los que la red pueda construir sus propias asociaciones. Se necesita, por tanto, una cantidad mucho mayor de patrones de entrada durante el entrenamiento para que la red pueda ajustar correctamente sus pesos sinápticos. Por supuesto, los procedimientos de aprendizaje son diferentes a los utilizados con el modelo de entrenamiento supervisado (Gestal 2013).

En este tipo de aprendizaje, lo que de hecho se está haciendo es exigirle a la red que capte por sí misma alguna de las características presentes en los datos de entrada. De hecho, en el aprendizaje no supervisado se pretende que las neuronas se autoorganicen aprendiendo a captar las regularidades de los datos de entrada sin suministrarles ningún tipo de criterio o ayuda externa que dirija dicha autoorganización (Gestal 2013).

c. Tipo de neuronas

Finalmente, también se pueden clasificar las RNA según sean capaces de procesar información de distinto tipo en:

- **Función de entrada.** Entre las más habituales se encuentra:

1) Sumatoria de las entradas pesadas: es la suma de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos.

$$\sum_j (in_n \cdot w_n) \quad \text{con } j = 1, 2, \dots, n$$

2) Productorio de las entradas pesadas: es el producto de todos los valores de entrada a la neurona, multiplicados por sus correspondientes pesos.

$$\prod_j (in_n \cdot w_n) \quad \text{con } j = 1, 2, \dots, n$$

3) Máximo de las entradas pesadas: sólo toma en consideración el valor de entrada más fuerte, previamente multiplicado por su peso correspondiente.

$$\text{Max}_j (in_n \cdot w_n) \quad \text{con } j = 1, 2, \dots, n$$

- **Función de activación.** Calcula el estado de actividad de una neurona. Procesan datos de entrada de naturaleza analógica, valores reales continuos, habitualmente acotados y usualmente en el compacto $[-1,1]$ o en el $[0,1]$, para dar respuestas también continuas. Las más habituales son: función lineal, función sigmoidea y función hiperbólica.
- **Función de salida.** Dos de las funciones de salida más comunes son:
 - 1) Ninguna: este es el tipo de función más sencillo, tal que la salida es la misma que la entrada. Es también llamada función identidad.
 - 2) Binaria: $Binaria \begin{cases} 1 & \text{si } f(x) \geq \xi_i ; \\ 0 & \text{de lo contrario} \end{cases}$; donde ξ_i es el umbral.

2.2.8. Tipos de redes neuronales artificiales

Se pueden distinguir varios tipos de RNA en función de la forma en que se organizan las neuronas, como aprenden o el número de capas. Aquí sólo se tratarán algunas de las más importantes y las que se consideran necesarias para el tema que concierne.

a. *Perceptrón simple*

En 1958 el psicólogo *Frank Rosenblatt* desarrolló un modelo simple de neurona basado en el modelo de *McCulloch* y *Pitts* y en una regla de aprendizaje basada en la corrección del error. A este modelo le llamó *perceptrón*. El *perceptrón* está constituido por conjunto de sensores de entrada que reciben los patrones de entrada a reconocer o clasificar y una neurona de salida que se ocupa de clasificar a los patrones de entrada en dos clases, según que la salida de la misma sea 1 (activada) o 0 (desactivada).

Suponer que se tiene una función $f(\cdot)$ de R^n en $\{-1,1\}$, que aplica un patrón de entrada $x = (x_1, x_2, \dots, x_N)^T \in R^n$ en la salida deseada $t \in \{-1,1\}$, es decir, $f(x) = t$. La información de que se dispone sobre dicha función viene dada por p pares de patrones de entrenamiento:

$$\{x_1, t_1\}, \{x_2, t_2\}, \dots, \{x_p, t_p\}$$

donde $x_i \in R^n$ y $f(x_i) = t_i \in \{-1,1\}$, $i = 1, 2, \dots, p$. Dicha función realiza una partición en el espacio R^n de patrones de entrada; por una parte estarían los patrones con salida +1 y por otra parte los patrones con salida -1. Por lo tanto, se puede decir que la función $f(\cdot)$

clasifica a los patrones de entrada en dos clases. Ejemplos de funciones $f(\cdot)$ de este tipo son la función lógica OR o la función par.

El *perceptrón* presenta dos capas de unidades procesadoras y sólo una de ellas presenta la capacidad de adaptar o modificar los pesos de las conexiones. La arquitectura del *perceptrón* admite capas adicionales, pero éstas no disponen la capacidad de modificar sus propias conexiones.

La Figura 13 muestra la unidad procesadora básica del *perceptrón*. Las entradas x_i llegan por la parte izquierda, y cada conexión con la neurona j tiene asignada un peso de valor w_{ji} .

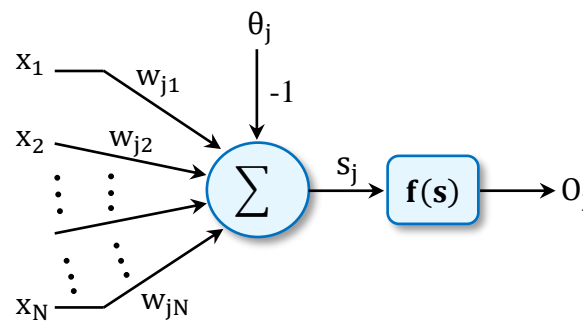


Figura 13: Perceptrón simple
FUENTE: Adaptado de [Sánchez \(2014\)](#)

La unidad procesadora del *perceptrón* realiza la suma ponderada de las entradas según la ecuación:

$$s_j = \sum x_i w_{ji}$$

Un aspecto común en muchas de las RNA es la entrada especial llamada “*bias*” representada en la figura por θ_j . Esta entrada siempre representa un valor fijo y funciona como una masa en un circuito eléctrico donde no varía de valor.

El *perceptrón* comprueba si la suma de las entradas ponderadas es mayor o menor que un cierto umbral Δ y genera la salida O_j según la siguiente ecuación:

$$f(s_j) = f(x_1, x_2, \dots, x_n) = \begin{cases} 1 & \text{si } w_{j1}x_1 + w_{j2}x_2 + \dots + w_{jN}x_N \geq \Delta \\ -1 & \text{si } w_{j1}x_1 + w_{j2}x_2 + \dots + w_{jN}x_N < \Delta \end{cases}$$

donde los parámetros $w_{j1}, w_{j2}, \dots, w_{jN}$ se llaman pesos sinápticos y son los pesos con los que se ponderan los valores de entrada x_1, x_2, \dots, x_n o argumentos de la función, y la suma ponderada $\mu = w_{j1}x_1 + w_{j2}x_2 + \dots + w_{jN}x_N$ es el potencial sináptico del que se ha

hablado anteriormente. También se puede expresar la función $f(\cdot)$ mediante la función signo, es decir,

$$f(x_1, x_2, \dots, x_n) = \text{sgn}(\mu - \Delta)$$

Siendo la función signo:

$$\text{sgn}(x) = \begin{cases} 1 & x \geq 0 \\ -1 & x < 0 \end{cases}$$

y se dirá que en este caso la función de transferencia es la función signo. Cuando la salida de la unidad es igual a 1 se dice que la unidad de proceso está activada y presenta el estado 1, mientras que si su salida es igual a cero se dice que está desactivada, presentando el estado 0 (Sánchez 2014).

b. *Perceptrón multicapa (MLP)*

El *perceptrón* multicapa (MLP por sus siglas en inglés) es una red de alimentación hacia adelante (*feedforward* en inglés) compuesta por una capa de unidades de entrada (sensores), otra capa de unidades de salida y un número determinado de capas intermedias de unidades de proceso, también llamadas capas ocultas porque no se ven las salidas de dichas neuronas y no tienen conexiones con el exterior. Cada sensor de entrada está conectado con las unidades de la segunda capa, y cada unidad de proceso de la segunda capa está conectada con las unidades de la primera capa y con las unidades de la tercera capa, así sucesivamente. Las unidades de salida están conectadas solamente con las unidades de la última capa oculta (Sánchez 2014), como se muestra en la Figura 14.

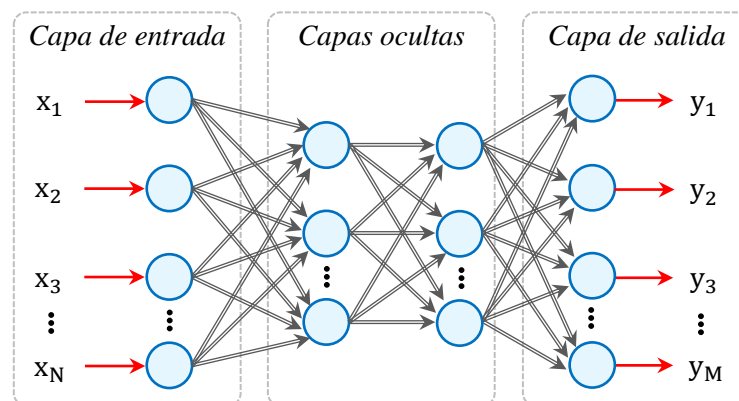


Figura 14: *Perceptrón multicapa (MLP)*

FUENTE: Adaptado de Sánchez (2014)

Con esta red se pretende establecer una correspondencia entre un conjunto de entrada y un conjunto de salidas deseadas, de manera que:

$$(x_1, x_2, \dots, x_N) \in \mathbb{R}^N \rightarrow (y_1, y_2, \dots, y_M) \in \mathbb{R}^M$$

Para ello se dispone de un conjunto de p patrones de entrenamiento, de manera que se sabe perfectamente la salida correspondiente a cada patrón de entrada. Así, el conjunto de entrenamiento será:

$$\{(x_1^k, x_2^k, \dots, x_N^k) \rightarrow (y_1^k, y_2^k, \dots, y_M^k): k = 1, 2, \dots, p\}$$

Para implementar dicha relación, la primera capa (sensores de entrada) tendrá tantos sensores como componentes tenga el patrón de entrada, es decir, N ; la capa de salida tendrá tantas unidades de proceso como componentes tengan las salidas deseadas, es decir, M , y el número de capas ocultas y su tamaño dependerán de la dificultad de la correspondencia a implementar.

Con esta arquitectura se consiguieron solventar las limitaciones de las redes de una sola capa, consiguiendo implementar cualquier función con el grado de precisión deseado, es decir, que las redes multicapa fuesen aproximadores universales. Por ejemplo, si se desea implementar la función XOR basta con utilizar dos unidades de proceso en la capa oculta, puesto que hay sólo dos patrones de entrada a los que les asignan una salida igual a 1. Al igual que las funciones booleanas, se puede implementar con esta red cualquier función continua de \mathbb{R}^N en \mathbb{R}^M , con la diferencia de que las entradas serán número reales (señales analógicas) y las unidades de proceso continuas (Sánchez 2014).

c. Redes de funciones de base radial (RBF)

Las funciones radiales son simplemente una clase de funciones. En principio, podían ser empleadas en cualquier tipo de modelo lineal o no lineal, y en cualquier tipo de red, monocapa o multicapa. Sin embargo, desde el estudio de *Broomhead* y *Lowe* en 1988, las redes de funciones de base radial (redes RBF) han sido asociadas tradicionalmente a redes compuestas por una sola capa oculta, tal y como muestra la Figura 15.

Como se observa en ella, cada vector de entrada x llega a m funciones base cuyas salidas se combinan linealmente con los pesos $\{w_j\}_{j=1}^m$ hacia la salida de la red $f(x)$.

Una red RBF es no lineal si las funciones base pueden cambiar de tamaño o si posee más de una capa oculta. El caso habitual es el de una red de una sola capa oculta cuyas funciones

están fijas en posición y tamaño. Además, suelen usarse algoritmos de optimización no lineal pero sólo para la regularización de los parámetros en el caso de regresión y para la elección del número óptimo de funciones base (que equivale a decir el número óptimo de nodos ocultos) en selección hacia delante.

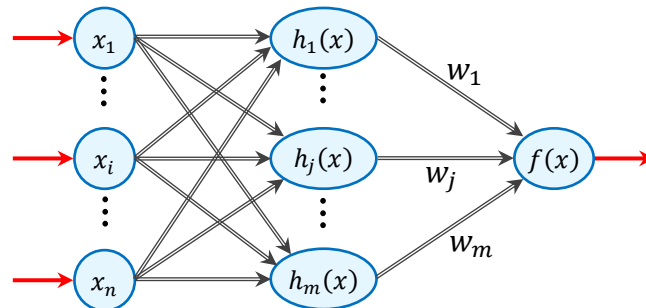


Figura 15: Red de función de base radial (RBF)
FUENTE: Adaptado de [Sánchez \(2014\)](#)

Por el contrario, en su entrenamiento se evitan el tipo de algoritmos no lineales de descenso por gradiente (tales como el gradiente conjugado o métodos de métrica variable) que son empleados en redes no lineales específicas, ya que manteniéndose firme en el mundo del álgebra lineal se consigue un análisis más sencillo y de menor costo computacional ([Orr 1996](#)).

Por lo tanto, las características principales de este tipo de redes son:

- Las redes de base radial tienen sus orígenes a finales de los 80.
- Son redes de tipo multicapa que tienen conexiones hacia delante y que sólo tienen una capa oculta. Cuyos nodos ocultos poseen carácter local, los nodos de salida realizan una combinación lineal de las activaciones de los nodos ocultos.
- Este tipo de redes construyen aproximaciones que son combinaciones lineales de múltiples funciones locales no lineales.
- Entre sus aplicaciones se encuentran el análisis de series temporales, procesamiento de imágenes, reconocimiento automático del habla, diagnósticos médicos, etc.

2.2.9. Perceptrón multicapa (MultiLayer Perceptrón)

Este tipo de red fue creada por *Werbos, Parker y Rumelhart* entre 1974 y 1986. Es una de las estructuras más utilizadas en el campo de la hidrología. Este tipo de red se compone, generalmente, de varias capas de nodos. La primera capa es la de entrada, donde se recibe la

información externa. Es transparente y sirve sólo para proveer de información a la red. La última, es la capa de salida, donde se obtiene el resultado del problema. Los nodos entre capas adyacentes, en general, se conectan totalmente de una capa inferior a una superior, ver la Figura 14.

El número de capas ocultas y la cantidad de nodos en cada capa se determinan usualmente por el método de prueba y error. Los nodos entre las capas adyacentes están totalmente conectados y se le asigna a cada conexión un peso sináptico que representa la fuerza relativa del nexo entre dos nodos o neuronas (Pujol 2009).

Antes de que una red desarrolle una determinada tarea, debe ser entrenada para hacerla. Básicamente, el entrenamiento es el proceso en el cual se determinan los pesos de las conexiones que son el elemento clave de una red. El conocimiento almacenado en una red está en las conexiones y nodos en forma de pesos y umbrales. Es precisamente a través de las conexiones, como una red puede realizar mapeos no lineales y muy complejos de los nodos de entrada a los nodos de salida. En este tipo de redes el entrenamiento es supervisado, es decir que la respuesta deseada de la red (los valores objetivos) para cada conjunto de entradas siempre está disponible.

Los datos de entrada utilizados para el entrenamiento se presentan en forma de vectores, donde cada elemento del vector de entrada corresponde a un nodo de entrada a la red. Por lo tanto, la cantidad de nodos de entrada a la red es igual a la dimensión del vector de entrada. Para un problema de pronóstico, el número de nodos de entrada está bien definido y corresponde a la cantidad de variables independientes asociadas al problema. En cambio, para un problema de pronóstico de series temporales, la cantidad apropiada de nodos de entrada no es de fácil determinación. Generalmente, los vectores de entrada para un problema de este tipo siempre están compuestos de una ventana de longitud fija que se mueve a lo largo de la serie. El total de datos disponibles se divide generalmente en dos o tres partes:

- **Serie de calibración:** Para determinar los pesos de las conexiones y los umbrales de los nodos.
- **Serie de validación:** Mide la habilidad de generalización de la red.
- **Serie de verificación:** Es opcional y se utiliza para verificar que la red no se sobre-entrene.

El proceso de entrenamiento es el siguiente: primero, se introducen en los nodos de entrada de la red, los ejemplos de la serie de entrenamiento. Los valores de activación de los nodos de entrada se ponderan y acumulan en cada nodo de la primera capa oculta. El total se transforma con la función de activación de la capa. Estos valores de activación, se convierten luego en las entradas a los nodos de la segunda capa oculta, y se repite este proceso hasta obtener los valores de activación de los nodos de salida (Pujol 2009).

El algoritmo de entrenamiento se utiliza para encontrar los pesos que minimicen una función objetivo, como puede ser la suma de los errores al cuadrado, entre otras. De ahí que el entrenamiento de una red es un problema de optimización no lineal sin restricciones.

Este tipo de arquitectura es estática, es decir que la red no tiene memoria para almacenar la información contextual del pasado. Por lo tanto, las entradas a la red deben contener toda la información contextual necesaria para representar las salidas (Koskela 2003).

La complejidad de este tipo de redes puede variar desde un modelo lineal hasta un modelo no lineal muy complejo, sólo variando la cantidad de capas y nodos y los valores de sus pesos. Una red de una sola capa oculta y \hat{y}_k salidas tiene la siguiente forma funcional:

$$\hat{y}_k = \eta + \sum_{h=1}^H \left\{ \beta_h \times g \left[\omega_h + \sum_{p=1}^P \alpha_{p,h} y_{k-p} \right] \right\} + e_k \quad (2.8)$$

donde P es el número de entradas, H el número de nodos en la capa oculta, $g(.)$ la función de activación de los nodos de la capa oculta, y_{k-p} son las entradas, y los pesos $\alpha_{p,h}$ y β_h corresponden a los nodos de la capa oculta y al nodo de salida, respectivamente.

Los algoritmos de entrenamiento suelen sobreentrenar las redes. El modelo con menor error en la etapa de entrenamiento, no siempre generaliza bien para datos nuevos. Por lo tanto, se deben utilizar métodos como el de parada temprana (*early stop* en inglés) para poder identificar estos problemas de entrenamiento.

A pesar de las buenas características de las RNA, construir una no es una tarea trivial. Una cuestión muy importante es determinar la arquitectura de la red, es decir, la cantidad de capas a considerar y sus respectivos nodos, así como la cantidad de conexiones entre ellos. Otras decisiones de diseño son: la elección de las funciones de activación de las capas ocultas y de salida, el algoritmo de entrenamiento, los métodos de transformación o normalización de los datos, la separación de los datos entre validación y calibración y las medidas de comportamiento. La elección de todos estos parámetros depende del problema.

Aunque existen varios métodos para la selección de la red óptima, son muy complejos y de difícil implementación y además ninguno de ellos puede garantizar que la red sea la óptima. Todo esto lleva a pensar que el diseño de una RNA sea más un arte que una ciencia (Zhang *et al.* 1998).

Este tipo de RNA puede ser utilizado para resolver una amplia variedad de problemas, como almacenamiento y búsqueda de información, aproximaciones funcionales, reconocimiento de patrones, filtrado de señales, segmentación de imágenes y señales, eliminación de ruido, control adaptativo, desarrollo de mapeos generales de patrones de entrada a patrones de salida o resolución de problemas de optimización (Pujol 2009).

Principales ventajas

- Capacidad de representación funcional universal.
- Gran rapidez de procesamiento.
- Ampliamente estudiada.
- Es la estructura conexionista que más se ha aplicado en la práctica.

Desventajas

- Tiempo de aprendizaje elevado para estructuras complejas.
- Necesidad de gran cantidad de datos de entrenamiento, para que pueda generalizar.
- Difícil determinación de la red óptima.

2.2.10. Algoritmos para la optimización (entrenamiento)

A través del tiempo, diversos métodos han sido propuestos en la literatura para el entrenamiento de la red neuronal, de las cuales resaltan las siguientes:

- Algoritmo de *Levenberg-Marquardt*.
- Técnicas clásicas de gradiente descendiente entre las que se incluyen la conocida Regla Delta Generalizada o algoritmo *Backpropagation*.
- *Resilient Backpropagation-Rprop*, *Rprop-iRprop* mejorada.

De los métodos enunciados se destacan las contribuciones de los algoritmos Regla Delta y *Levenberg-Marquardt*, por su uso extensivo en la literatura y *RProp*, por las bondades de sus resultados; los tres algoritmos se describen a continuación:

a. Algoritmo de *Levenberg-Marquardt*

Es un método iterativo de optimización, que utiliza de forma similar a los algoritmos de *quasi-Newton* (método de la secante) un gradiente de segundo orden, expresado en una aproximación de la matriz *Hessiana*, con lo cual se logra incrementar considerablemente la velocidad del entrenamiento.

Cuando la función de evaluación del comportamiento de la red es la suma cuadrática del error, la matriz *Hessiana* puede ser aproximada como:

$$H = J^T \cdot J \quad (2.9)$$

Y el gradiente puede obtenerse por:

$$g = J^T \cdot \varepsilon \quad (2.10)$$

donde J es la matriz *Jacobiana*, la cual contiene las primeras derivadas de los errores de la red con respecto a los pesos y al *bias*, y ε es el vector de errores. Esta aproximación permite un ahorro computacional importante en la determinación de la matriz *Hessiana*.

La matriz *Jacobiana* para un nodo se escribe como:

$$J = \begin{bmatrix} \frac{\partial \varepsilon_1}{\partial w_1} & \dots & \frac{\partial \varepsilon_1}{\partial w_n} & \frac{\partial \varepsilon_1}{\partial w_0} \\ \cdot & & \cdot & \cdot \\ \cdot & & \cdot & \cdot \\ \cdot & & \cdot & \cdot \\ \frac{\partial \varepsilon_p}{\partial w_1} & \dots & \frac{\partial \varepsilon_p}{\partial w_n} & \frac{\partial \varepsilon_p}{\partial w_0} \end{bmatrix} \quad (2.11)$$

El método determina la nueva configuración de pesos mediante:

$$w_{k+1} = w_k - [J^T \cdot J + \mu \cdot I]^{-1} \cdot J^T \cdot \varepsilon \quad (2.12)$$

donde: w es el vector de pesos, w_0 es el *bias* y ε es el vector de errores, μ es un escalar que cuando es igual a cero, la ecuación 2.12 se convierte en la del método de *Newton* modificado, utilizando la aproximación 2.9 para el cálculo de la matriz *Hessiana*.

Cuando μ se incrementa, el algoritmo es similar al método de descenso de gradiente, es decir de lenta aproximación, por lo que el objetivo es reducir el valor de μ para tender al método de *Newton*. El método considera inicialmente asignar un valor pequeño para μ y generalmente se asigna $\mu = 0.001$, si el valor de la función de minimización disminuye, se acepta la disminución del error y el valor de μ se disminuye, si por el contrario el error aumenta μ se incrementa y se evalúa nuevamente la ecuación 2.12. Este procedimiento se

realiza repetidamente haciendo variar los valores de μ hasta llegar a un valor aceptable del error (Mateo 2012).

El método de *Levenberg-Marquardt* es uno de los de mayor rapidez de aprendizaje y está recomendado para redes de un tamaño moderado de nodos. El principal inconveniente del método de *Levenberg-Marquardt* es el elevado requerimiento de memoria para su implementación (Adamowski y Sun 2010), debido a que las dimensiones de la matriz *Jacobiana* son de orden $m \times n$ donde m es el número de muestras utilizadas en el entrenamiento y n (pesos y *bias*) es el número total de parámetros de la red.

b. Algoritmo de regla delta generalizada o *Backpropagation*

Este método de aprendizaje supervisado para el entrenamiento de redes neuronales consiste en la minimización del error de ajuste del modelo de red a la serie de tiempo modificando los pesos en la dirección descendente de la función del error.

La retropropagación (*Backpropagation* en inglés), es uno de los algoritmos más robustos para el aprendizaje supervisado de redes multicapa *feedforward* (literalmente alimentación hacia adelante). Básicamente, este algoritmo utiliza repetidamente la regla de la cadena para establecer la influencia de cada peso w de la red respecto a una determinada función de error E :

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial a_i} \cdot \frac{\partial a_i}{\partial net_i} \cdot \frac{\partial net_i}{\partial w_{ij}} \quad (2.13)$$

donde w_{ij} es el peso de la neurona i a la neurona j , a_i es el valor de activación y net_i es la sumatoria de los pesos de las entradas de la neurona i . Una vez que la derivada parcial de cada peso es conocida, el resultado de minimizar la función del error es almacenado, y luego se realiza un descenso de gradiente simple:

$$w_{ij}(t + 1) = w_{ij}(t) - \varepsilon \frac{\partial E}{\partial w_{ij}}(t) \quad (2.14)$$

donde ε es el factor o tasa de aprendizaje, de ésta depende en mayor parte que el algoritmo converja. Consecuentemente, entre más pequeño se asigne ε más pasos son necesarios para que la técnica converja (Sánchez 2012).

c. Algoritmo *Rprop* (*Resilient Backpropagation*)

Rprop es una técnica basada en gradiente desarrollada por *Riedmiller* en 1994, citado por [Sánchez \(2012\)](#). Del conjunto de técnicas desarrolladas en la literatura basadas en el gradiente, ésta es la que tradicionalmente se usa para estimar los parámetros de una RNA. El algoritmo *Rprop*, difiere de la técnica clásica de *Backpropagation* en que las derivadas parciales de la función del error de ajuste sólo son usadas para determinar el sentido en que deben ser corregidos los pesos de la red, pero no las magnitudes de los ajustes. Los algoritmos basados en el gradiente descendente modifican los valores de los parámetros proporcionalmente al gradiente de la función de error, de tal forma que en regiones donde el gradiente tiende a ser plano el algoritmo avanza lentamente. *Rprop* no se ve afectado por la saturación de las neuronas de la red neuronal, ya que solamente se usa la derivada para determinar la dirección en la actualización de pesos. Consecuentemente, converge más rápidamente que los algoritmos basados en *Backpropagation* ([Sánchez 2012](#)).

La optimización de los parámetros del modelo (η , ω_h , β_h , $\alpha_{p,h}$ para $h = 1, \dots, H$ y $p = 1, \dots, P$ en la Ecuación 2.8), depende del grado de error existente entre el valor deseado y el pronosticado por la red durante el entrenamiento, y está asociado a su vez a la configuración de la red neuronal; es así como, la selección del número óptimo de variables del modelo está directamente relacionada con el proceso de entrenamiento usado. Tal como lo exponen [Qi y Zhang \(2001\)](#), existe una estrecha relación entre el desempeño del modelo y la selección de los valores de P y H . La importancia primaria de hacer una selección adecuada radica en las dificultades de convergencia del algoritmo de aprendizaje, que puede acarrear el incluir retardos irrelevantes y obtener un modelo final con pobre desempeño en ajuste y generalización ([Sánchez 2012](#)).

En conclusión, para un MLP, y bajo un esquema constructivo de adicción de neuronas ocultas, debería presentarse una reducción sistemática del error de ajuste cada vez que se agrega una neurona en la capa oculta, hasta un nivel de precisión del ajuste arbitrariamente cercano a cero ([Sánchez y Velásquez 2010](#)).

2.2.11. Aprendizaje y generalización de las RNA

Una característica importante de las RNA es su capacidad para aprender y generalizar. El proceso de aprendizaje (entrenamiento) de una red neuronal se logra ajustando los pesos

asociados con las conexiones entre las neuronas. Por lo tanto, las RNA adolecen de un problema común en los métodos de aprendizaje supervisado denominado sobreaprendizaje o sobreajuste (*overfitting*). El sobreaprendizaje ocurre cuando una red neuronal ha aprendido el modelo correctamente en la etapa de entrenamiento, pero no responde adecuadamente a la validación. Esto sucede por diversas causas como, por ejemplo, porque el número de ciclos de aprendizaje es muy elevado, o porque el número de neuronas de la capa oculta es también muy elevado, etc.

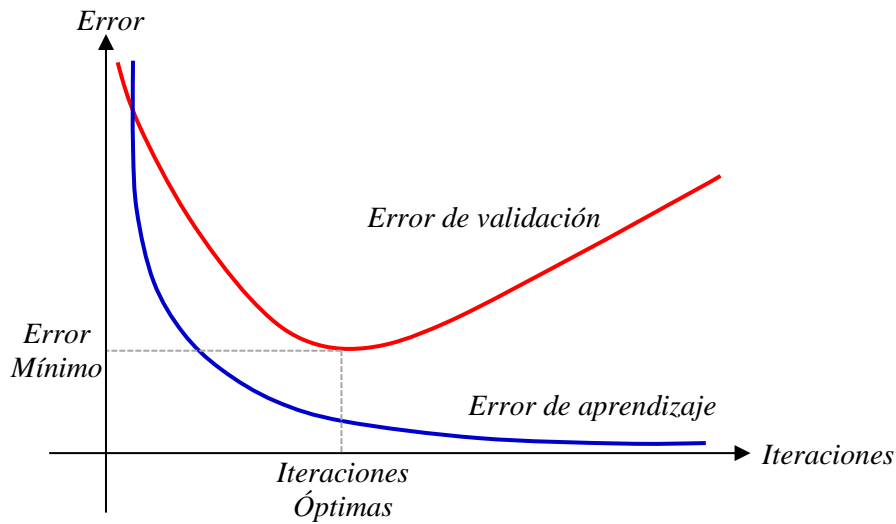


Figura 16: Sobreajuste o sobreaprendizaje (*overfitting*) en una red neuronal
FUENTE: Adaptado de [Gallardo \(2009\)](#) y [Morales \(2010\)](#)

El mejor modo de evitar el sobreaprendizaje es lograr un mayor número de casos para el entrenamiento. Cuando esto no sea posible también se puede limitar el número de nodos en la capa oculta, limitar el número de ciclos de aprendizaje o usar la técnica de validación cruzada (*cross-validation*). En la Figura 16 que muestra un caso de red neuronal sobreentrenada, el error de entrenamiento y validación se muestran en azul y rojo, respectivamente. Si el error de validación aumenta mientras que el de entrenamiento decrece puede que se esté produciendo una situación de sobreajuste ([Gallardo 2009](#)).

2.2.12. Ventajas y limitaciones de las RNA

Son muchas las ventajas y limitaciones que se podrían argumentar a favor y en contra de las RNA. Sin embargo, lo más adecuado sería analizar individualmente cada tipo de red, pero ese no es el objetivo de esta sección ni de este documento ([Pujol 2009](#); [Espinoza 2010](#); [Rivera 2005](#)).

Ventajas

- La ventaja más importante de las RNA se encuentra en la resolución de problemas que son demasiado complejos para las tecnologías convencionales, problemas que no tienen una solución algorítmica o problemas en los que la solución algorítmica es demasiado complicada para ser identificada
- Proporcionan una alternativa analítica a las técnicas convencionales que frecuentemente están limitadas por estrictas hipótesis de normalidad, linealidad, independencia, etc.
- Por su facilidad para capturar muchas clases de relaciones, permiten al usuario de una forma relativamente fácil modelar fenómenos que con otras metodologías podría resultar muy complicado o incluso imposible.

Limitaciones

- Las teorías matemáticas usadas para garantizar el desempeño de las RNA aplicadas están aún en desarrollo.
- Inestabilidad para explicar los resultados que obtienen.
- Problemas de escalabilidad, prueba y verificación.
- El producto final de la actividad de una RNA es una red entrenada que no proporciona ecuaciones o coeficientes que definan una relación (como en los modelos de regresión) que vaya más allá de su propia matemática interna. La red es la ecuación final de la relación.

2.2.13. Pronóstico de series de tiempo utilizando RNA

Es indiscutible que los modelos multivariados brindan mayores ventajas que los modelos univariados para la comprensión de la estructura subyacente de un sistema dado y en ocasiones en la obtención de mejores pronósticos. Pero, no siempre esta aseveración es cierta, debido a que, al existir más parámetros por estimar hay más oportunidades de variación muestral que incrementa la incertidumbre y afecta los pronósticos. Sin embargo, los modelos univariados suelen ser útiles para muchos fines incluyendo el pronóstico hidrológico.

a. Predictor individual

La RNA es una caja negra que tiene una o más entradas y exactamente una salida por lo general en hidrología. La estructura más simple posible se consigue utilizando sólo rezagos de una variable en particular como entradas a la red y tratando de predecir un punto adelante en una sola serie de tiempo.

Un esquema típico de una serie temporal a una red neuronal para el pronóstico se describe en la Figura 17. Este esquema muestra el uso de seis puntos continuos (de una sola variable) en una serie de tiempo para pronosticar el siguiente punto. La serie de entrenamiento se produce mediante la generación de un gran número de muestras individuales. Cada muestra se compone de siete puntos: el punto actual (rezago 0), cinco puntos históricos (rezagos de 1 a 5), y el punto siguiente (a menudo llamada la salida objetivo), que se utiliza para dirigir el entrenamiento de la neurona de salida. El conjunto de entrenamiento puede llegar a ser muy grande.

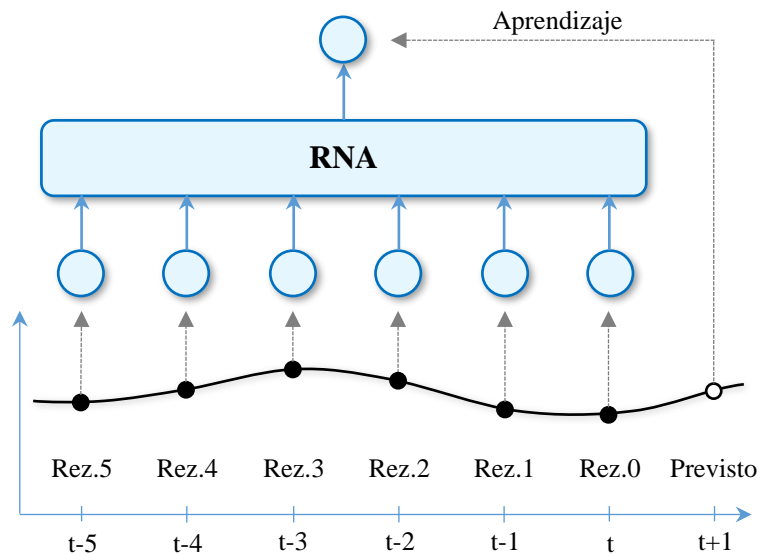


Figura 17: Modelo de pronóstico de series de tiempo univariado

FUENTE: Adaptado de Zealand (1997)

La Figura 17 muestra el pronóstico exactamente un punto en el futuro. Muchas veces, se necesitan prever más alejado que sólo un punto. Múltiples pronósticos a partir de una red son posibles, pero por lo general es mejor evitarlo. Lo mejor es limitar una red para tener que hacer sólo un pronóstico de anticipación, ya que la anticipación es un punto en el futuro o cien puntos en el futuro. Aunque la mayoría de las redes pueden acomodarse fácilmente a múltiples salidas, es mejor no intentar formar múltiples salidas para entrenar

distintos pronósticos de anticipación. En su lugar, entrenar las redes separadas para cada tiempo de anticipación (Zealand 1997).

b. Series de predictores múltiples

En los pronósticos de series temporales más básicas se usan rezagos a partir de una serie temporal de una sola variable como los predictores, como se discutió en la sección anterior. Si es necesario, es posible hacer uso de más de una variable, junto con sus retrasos también, así extendiéndose a múltiples series temporales de múltiples variables como puede verse en la Figura 18.

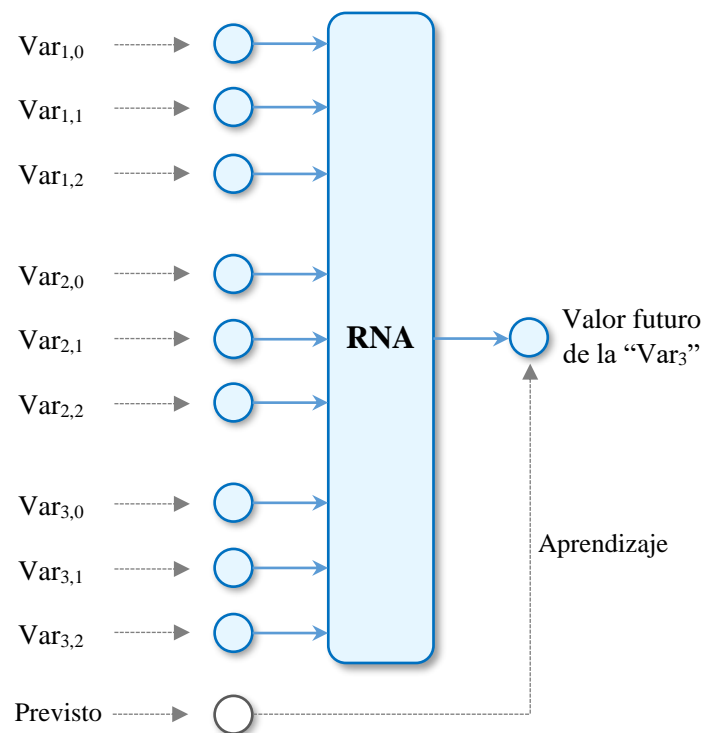


Figura 18: Modelo de pronóstico de series de tiempo multivariado

FUENTE: Adaptado de Zealand (1997)

La Figura 18 muestra el uso de tres rezagos de tres variables diferentes, Var_{ij} , donde i corresponde al tipo variable y j corresponde al rezago de dicho tipo de variable. Los rezagos de estas tres variables crean tres series de tiempo independiente para predecir el siguiente punto de la Var_3 . Una vez más, la serie de entrenamiento se produce mediante la generación de un gran número de muestras individuales. En este caso, cada muestra se compone de diez puntos (9 entradas y 1 salida). Las entradas son el punto actual (rezago 0) de los tres tipos de variables y dos puntos históricos (rezagos de 1 y 2) de los tres tipos

de variables. La salida es el punto siguiente (a menudo llamada la salida objetivo) de la variable que se previó, que se utiliza para dirigir el entrenamiento de la neurona de salida (Zealand 1997).

2.2.14. Aplicaciones de RNA para el pronóstico de caudales en el Perú

En los últimos años, se han realizado reducidas aplicaciones de redes neuronales artificiales (RNA) para el pronóstico de caudales diarios/mensuales en las cuencas hidrográficas peruanas y amazónicas. En seguida, se describen algunos trabajos que han sido publicadas en revistas de investigación reconocidas.

Laqui (2010) desarrolló modelos basados en redes neuronales artificiales para la previsión de los caudales medios mensuales del río Huancané, utilizando como datos de entrada caudales, precipitación y evaporación. Así, concluyó que los modelos de RNA mostraron un excelente desempeño en términos de medidas de error, frente a un modelo autorregresivo periódico de primer orden - PAR(1), ya que presentó un porcentaje de ajuste de 72 por ciento, en comparación al 59 por ciento que presentó el modelo de serie de tiempo estocástica.

Lujano *et al.* (2014) realizaron pronósticos de caudales medios mensuales del río Ilave, utilizando modelos de redes neuronales artificiales, y empleando datos de caudales y precipitación. Los resultados de validación y prueba indicaron calificativos de buenos a muy buenos, con coeficientes de *Nash-Sutcliffe* superiores a 87 por ciento y raíz del error medio cuadrático inferiores a 20 por ciento, mostrando un buen desempeño en la estimación de fenómenos de comportamiento no lineal como los caudales.

Béjar *et al.* (2016) aplicaron redes neuronales artificiales y modelos neurodifuso ANFIS para la predicción de caudales medios diarios en una sección de la cuenca del Amazonas, usando caudales de las estaciones de Obidos, Fazenda Vista Alegre, Manacapuru y Serrinha, y datos de precipitación de una estación ubicada muy cerca de la desembocadura del río Branco. De la evaluación del desempeño de ambos modelos, mostraron que las técnicas de la inteligencia artificial consiguieron coeficientes de correlación superiores al 97 por ciento y errores medio porcentual absoluto por debajo del 10 por ciento, los cuales, fueron calificados como buenos para la predicción, gracias a su capacidad predictiva en comparación con métodos tradicionales del tipo lineal.

2.3. DESCRIPCIÓN DE LA TRANSFORMADA DE WAVELET

Muchos de los fenómenos físicos pueden describirse mediante una señal en el dominio del tiempo; es decir, una de las variables es el tiempo (variable independiente) y la otra la amplitud (variable dependiente). Cuando se dibuja esta señal se obtiene una función tiempo-amplitud; sin embargo, la información que se puede obtener directamente de esta representación no siempre es la más apropiada, puesto que la información que caracteriza a la señal, en muchos casos, puede observarse más claramente en el dominio de la frecuencia, es decir, mediante un espectro de frecuencias que muestre las frecuencias existentes en la señal. Por lo tanto, para una mejor representación de la señal se hace necesario disponer de su representación en el dominio del tiempo y de la frecuencia.

2.3.1. Introducción a las transformadas

Tradicionalmente las señales (o funciones) se representan en el dominio del tiempo (o del espacio), ya sea en forma de funciones continuas (señales analógicas) o como muestreos discretos en el tiempo (señales digitales). La transformación desde el dominio del tiempo al dominio de la frecuencia revela otros aspectos de las señales como la magnitud de cambio de los componentes de la señal en el tiempo o el espacio. La transformada de *Fourier* (TF) es una herramienta con la capacidad de representar señales o funciones tanto en el dominio del tiempo como en el dominio de la frecuencia. El análisis de *Fourier* ha dominado el procesamiento y análisis de señales en el último medio siglo, debido principalmente a que la frecuencia es la llave para el procesamiento y filtrado de señales físicas.

Sin embargo, ciertas señales, cuya amplitud varía en forma rápida y abrupta en el tiempo o señales cuyo contenido de frecuencia es variable de un instante de tiempo a otro (señales no estacionarias) no pueden ser analizadas a fondo mediante la TF, debido a ciertas limitaciones de este análisis en el campo tiempo-frecuencia. La TF detecta la presencia de una determinada frecuencia, pero no brinda información acerca de la evolución en el tiempo de las características espectrales de la señal. Muchos aspectos temporales de la señal, tales como el comienzo y el fin de una señal finita y el instante de aparición de una singularidad en una señal transitoria, no pueden ser analizados adecuadamente por el análisis de *Fourier*. Es en estos términos de análisis donde entra en juego una nueva herramienta matemática llamada transformada de *wavelet* (TW).

El término *wavelet* se define como una “ondeleta” o función localizable en el tiempo. Es decir, las *wavelets* son funciones matemáticas que proporcionan una representación en tiempo y frecuencia de las series temporales estacionarias como no estacionarias (Adamowski 2008; Dadu y Deka 2013). Así, vista desde una perspectiva del análisis o procesamiento de señales puede ser considerada como una herramienta matemática para la representación y segmentación de señales, análisis tiempo-frecuencia e implementación de algoritmos sencillos y rápidos desde el punto de vista computacional. Las características propias de la TW otorgan la posibilidad de representar señales en diferentes niveles de resolución, representar en forma eficiente señales con variaciones de picos abruptos, así como analizar señales no estacionarias. Permite conocer el contenido en frecuencia de una señal y cuándo estas componentes de frecuencia se encuentran presentes en la señal (Acevedo 2009). Otra ventaja del análisis TW es la elección flexible de la *wavelet* madre de acuerdo con las características de las señales investigadas (Adamowski 2008).

2.3.2. La transformada de *Fourier*

La transformada de *Fourier* (TF), en esencia, descompone o expande una señal o función en senos y cosenos de diferentes frecuencias cuya suma corresponde a la señal original. En otras palabras, es capaz de distinguir las diferentes componentes de frecuencia de la señal, y sus respectivas amplitudes. Puede verse como una técnica matemática para transformar el punto de vista de una señal desde la base de tiempo a la base de la frecuencia, tal como se representa esquemáticamente en la Figura 19.

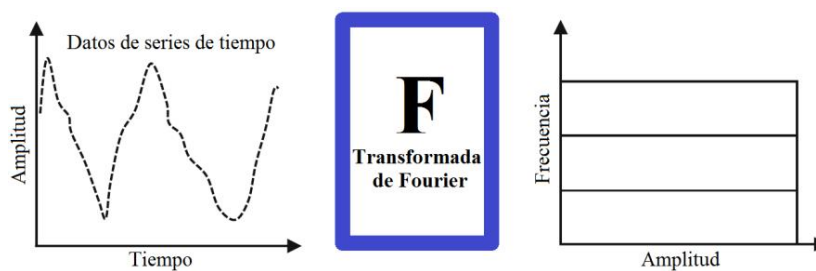


Figura 19: Esquema de transformación de la transformada de *Fourier*

FUENTE: Adaptado de Acevedo (2009)

La TF de una función del tiempo $f(t)$ se define como:

$$F(\omega) = \int_{-\infty}^{+\infty} f(t)e^{i\omega t} dt \quad (2.15)$$

y la transformada inversa de *Fourier*, como:

$$f(t) = \int_{-\infty}^{+\infty} J(\omega)e^{-i\omega t}d\omega \quad (2.16)$$

La TF puede obtener una representación en el dominio de la frecuencia de una señal que se encuentra originalmente en el dominio del tiempo. La relación existente entre la representación de la señal original a través de funciones senoidales y cosenoidales y la exponencial, proviene de la definición de la identidad de *Euler*:

$$e^{i\omega t} = \cos(\omega t) + \text{sen}(\omega t)i \quad (2.17)$$

$$e^{-i\omega t} = \cos(\omega t) - \text{sen}(\omega t)i \quad (2.18)$$

Mediante esta función exponencial es posible formar un conjunto de funciones ortogonales de la forma:

$$\{e^{in\omega t} : n = 0, \pm 1, \pm 2, \pm 3, \dots\} \quad (2.19)$$

sobre un intervalo $[t_0, t_0+T]$ y por lo tanto se puede descomponer o expandir la señal original (en el dominio del tiempo) de la siguiente manera:

$$f(t) = \sum_{-\infty}^{+\infty} F_n e^{-in\omega t} \quad (2.20)$$

Estas funciones exponenciales pueden ser referidas como las funciones bases de la TF, y debido a su propiedad de ortogonalidad, es posible obtener los coeficientes F_n como términos de semejanza entre la señal original y la función exponencial (Acevedo 2009).

$$F_n = \frac{1}{T} \int_{t_0}^{t_0+T} f(t)e^{in\omega t} dt \quad (2.21)$$

La TF es una herramienta muy útil para el análisis de señales estacionarias y convertir la función compleja en el dominio de frecuencia simple. Pero la TF no es capaz de analizar las señales no estacionarias o proporcionar el periodo de tiempo de las señales (Labat 2005).

2.3.3. Transformada de *wavelet*

El análisis *wavelet* representa el paso lógico siguiente a la transformada corta de *Fourier* (una técnica mediante ventanas con regiones de tamaño variable). El análisis *wavelet* permite el uso de intervalos grandes de tiempo en aquellos segmentos en los que se requiere mayor precisión en baja frecuencia, y regiones más pequeñas donde se requiere información en alta frecuencia. El esquema se muestra en la Figura 20.

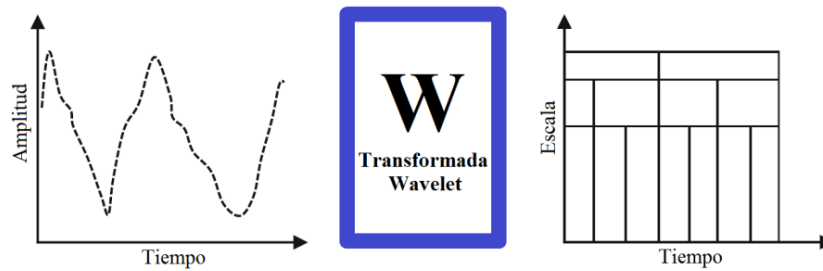


Figura 20: Esquema de transformación de la transformada de *wavelet*
FUENTE: Adaptado de [Acevedo \(2009\)](#)

Las *wavelets* son una clase de funciones que se utilizan para localizar una función dada tanto en posición y escala. Se utilizan en aplicaciones tales como procesamiento de señales y análisis de series de tiempo. Las *wavelets* forman la base de la transformada *wavelet* que "divide datos, funciones u operadores en diferentes componentes de frecuencia, y luego estudia cada componente con una resolución adaptada a su escala". En el contexto de procesamiento de señales, la transformada de *wavelet* depende de dos variables: escala (o frecuencia) y tiempo ([Veitch 2005](#)).

Hay dos tipos principales de la transformada de *wavelet*: continua y discreto. El primero está diseñado para trabajar con funciones definidas sobre todo el eje real. La segunda trata de las funciones que se definen en un rango de números enteros (usualmente $t = 0, 1, \dots, N - 1$, donde N representa el número de valores en la serie temporal).

a. Transformada de *wavelet* continua

La transformada de *wavelet* continua (TWC) intenta expresar una señal $x(t)$ continua en el tiempo, mediante una expansión de términos o coeficientes proporcionales al producto interno entre la señal y diferentes versiones escaladas y trasladadas de una función prototipo $\psi(t)$ más conocida como *wavelet* madre. Asumiendo que tanto la señal como la nueva función $\psi(t)$ son de energía finita, entonces se puede definir:

$$TWC(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) \psi\left(\frac{t-b}{a}\right) dt \quad (2.22)$$

Como se puede observar han aparecido dos nuevas variables a y b . Figura 21 es una ilustración de las formas desplazada y escalada de una *wavelet* madre. La variable a controla el ancho o soporte efectivo de la función $\psi(t)$, y la variable b da la ubicación en el dominio del tiempo de ψ .

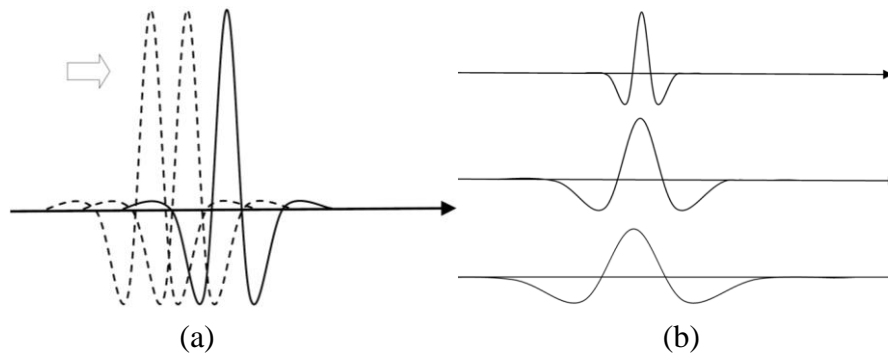


Figura 21: Ilustración de *wavelet* (a) desplazado y (b) escalado
FUENTE: Extraído de [Badrzadeh \(2014\)](#)

Ahora bien, para que este análisis sea posible y además para poder lograr una perfecta reconstrucción de la señal a partir de la transformada, la función $\psi(t)$ debe cumplir con la condición de admisibilidad. El cumplimiento de esta condición significa que el valor medio de $\psi(t)$ es igual a 0, lo que a su vez implica obligatoriamente que $\psi(t)$ tenga valores tanto positivos como negativos, es decir, que sea una onda. Además, como $\psi(t)$ es una función que “ventaniza” la señal sobre un intervalo de tiempo dado por a alrededor de un punto $t = b$, se observa intuitivamente que $\psi(t)$ es de soporte compacto, es decir, es una onda definida sobre un intervalo de tiempo finito, de ahí el porqué de su nombre, *wavelet* u “ondita”.

Mediante la variable de escala se puede comprimir ($|a| < 1$) o dilatar ($|a| > 1$) la función $\psi(t)$, lo que dará el grado de resolución con el cual se esté analizando la señal. Por definición la TWC es más una representación tiempo-escala que una representación tiempo-frecuencia. En particular, para valores pequeños de a la TWC obtiene información de $x(t)$ que está esencialmente localizada en el dominio del tiempo mientras que para valores grandes de a la TWC obtiene información de $x(t)$ que está localizada en el dominio de la frecuencia. Es decir, para escalas pequeñas la TWC entrega una buena resolución en el dominio del tiempo, mientras que para escalas grandes la TWC entrega una buena resolución en el dominio de la frecuencia. Cuando el valor de la variable a cambia, tanto la duración como el ancho de banda de la *wavelet* cambian, pero su forma se mantiene igual. En lo anteriormente dicho se encuentra la diferencia principal entre la TWC y la TF, ya que la primera ocupa ventanas de corta duración para altas frecuencias y ventanas de larga duración para bajas frecuencias, mientras que la TF ocupa una sola ventana con la misma duración tanto para altas frecuencias como para bajas frecuencias.

La TWC es una transformada reversible siempre y cuando la función $x(t)$ cumpla con la condición de admisibilidad. La reconstrucción es posible usando la siguiente fórmula (Acevedo 2009):

$$x(t) = \frac{1}{c_\psi^2} \int_a \int_b CWT_x^\psi(a, b) \frac{1}{a^2} \psi\left(\frac{t-b}{a}\right) db da \quad (2.23)$$

b. Transformada de *wavelet* discreta

El cálculo de TWC requiere una cantidad significativa de tiempo de cómputo y recursos, ya que la señal se analiza sobre un número infinito de dilataciones y traslaciones de la *wavelet* madre (Badrzadeh 2014). Teniendo en cuenta la naturaleza discreta de las series de tiempo hidrológicas observadas, se prefiere la transformada *wavelet* discreta (TWD) en la mayoría de los pronósticos hidrológicos (Sang 2012).

En la TWD, los parámetros de escala y traslación generalmente se basan en potencias de dos (diádico) en lugar de todas las escalas y traslaciones posibles. Por lo tanto, la ecuación de la TWC se puede escribir de la siguiente manera:

$$TWD(a, b) = \frac{1}{\sqrt{a}} \sum_{t=1}^{N-1} x(t) \psi\left(\frac{t-b}{a}\right) \quad (2.24)$$

donde N es la longitud de la señal discreta y $TWD(a, b)$ es el coeficiente *wavelet* para la *wavelet* discreta de escala $a = 2^j$ y traslación $b = k2^j$.

Los enteros j y k controlan la dilatación y la traslación de la *wavelet* madre $\psi(t)$, respectivamente. Sustituyendo a y b , la ecuación se convierte en:

$$TWD(j, k) = 2^{-j/2} \sum_{t=1}^{N-1} x(t) \psi(2^{-j}t - n) \quad (2.25)$$

La TWD opera sobre una función muestreado discretamente o series temporales $x(t)$, por lo general definiendo el tiempo $t = 0, 1, \dots, N-1$ para ser finito (Veitch 2005; Badrzadeh 2014).

2.3.4. Familia de funciones *wavelet*

Hay diferentes tipos de funciones *wavelets* (*wavelet* madre). En la ingeniería de recursos hídricos, la característica más importante de cada función *wavelet* es su forma. La selección apropiada de la función *wavelet* es la principal preocupación de modelamiento

híbrido *wavelet* en hidrología. La mejor sugerencia es el procedimiento de prueba y error mediante la aplicación de diferentes tipos de funciones *wavelets*. Tal vez, la mejor selección podría basarse en la similitud entre la forma de series temporales de caudal y sus coeficientes *wavelet* asociadas. Algunas de las funciones *wavelets* más importantes son *Haar*, *Daubechies* (db), *Mexican-Hat*, *Symlets*, *Coiflets* (Coifmann), *Morlet*, etc. (Badrzadeh 2014).

Wavelet Daubechies (db), la más popular función *wavelet*, fue introducido por primera vez por Ingrid Daubechies. La familia *Daubechies* es ortogonal y asimétrica. Esta función *wavelet* se presenta a menudo por dbN (donde: db es el “apellido” y N es el orden de la *wavelet* aplicada). La Figura 22 presenta algunas muestras de familias de *Daubechies*.

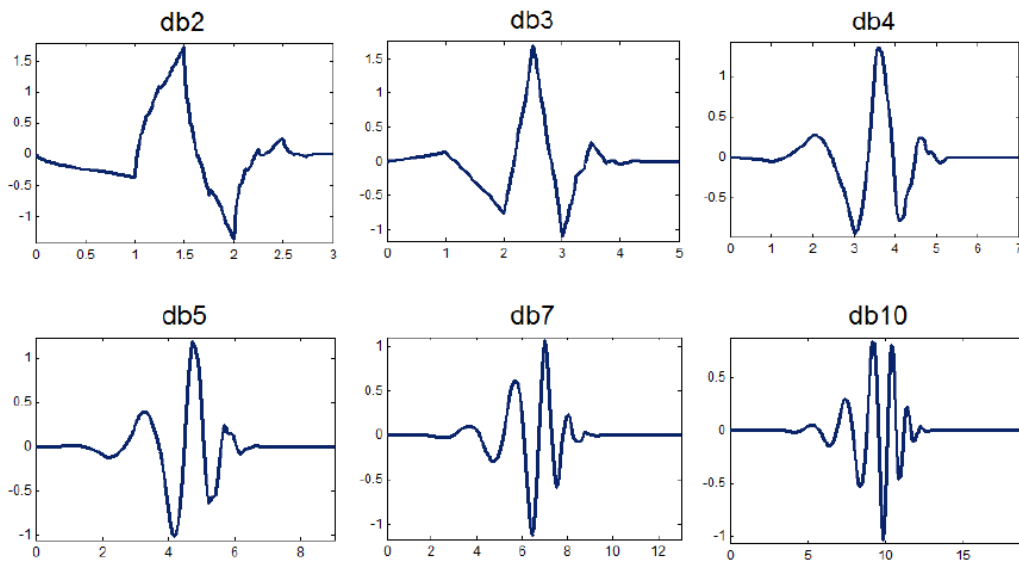


Figura 22: Familia de *wavelet Daubechies*
FUENTE: Extraído de Badrzadeh (2014)

Prácticamente, el orden de *wavelet Daubechies* es que van desde uno hasta veinte. Estos órdenes representan el número de momentos de desvanecimiento de la función *wavelet*. Sin embargo, *Daubechies* orden uno (db1) es una distinción de otro orden de *Daubechies*. Se llama función *wavelet Haar*, la primera y más sencilla *wavelet*. La simple función de *wavelet Haar* es el siguiente;

$$\psi_t \begin{cases} 1 & \text{si } 0 \leq t < 0.5 \\ -1 & \text{si } 0.5 \leq t < 1 \\ 0 & \text{otros valores de } t \end{cases} \quad (2.26)$$

Esta simplicidad *wavelet* tiene la ventaja de ser rápido y eficiente de la memoria. La Figura 23 ilustra la función *wavelet Haar*.

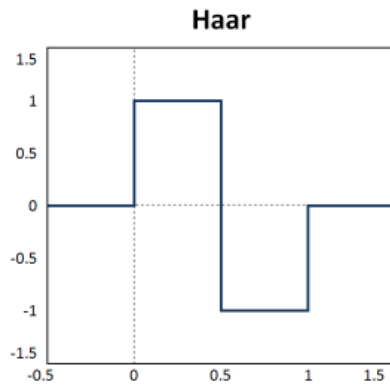


Figura 23: Función *wavelet* Haar
FUENTE: Extraído de [Badrzadeh \(2014\)](#)

Para mejorar la característica asimétrica de *wavelet Daubechies*, *wavelet Symlets* fue propuesto por *Daubechies*. El número de momentos de desvanecimiento y el tamaño de *wavelet Symlet* son similares a las *wavelets* db, pero tiene más simetría.

Otra *wavelet* popular es el *Coiflets*, diseñados por *Dubechies*. Esta *wavelet* lleva el nombre de *Ronald Coifman* por sus *wavelets* propuestos con igual número de momentos de desvanecimiento y funciones de escala. La Figura 24 presenta la forma de cuatro funciones *wavelet* diferentes.

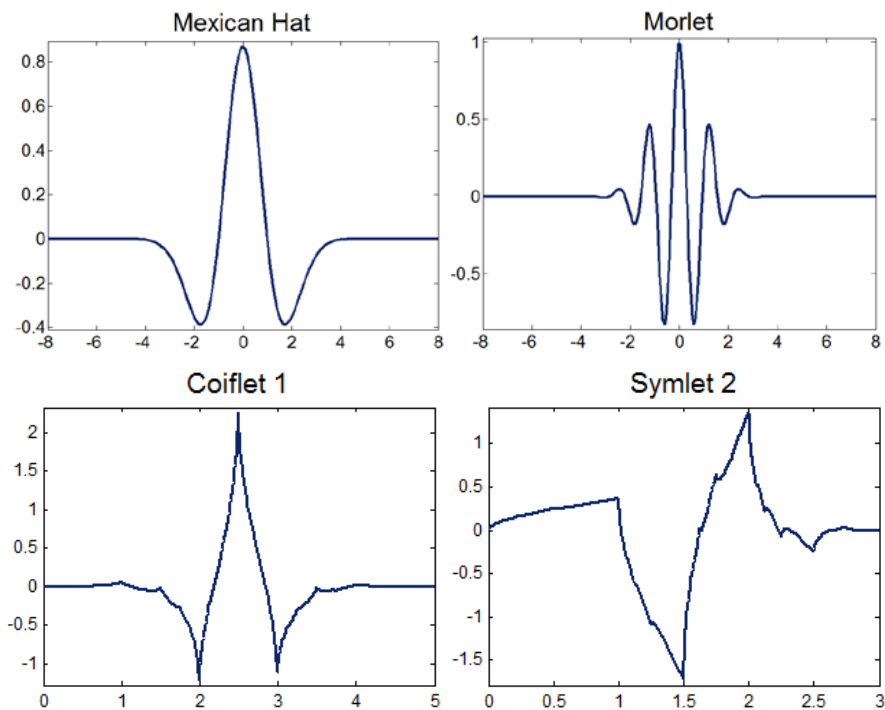


Figura 24: Funciones *wavelet* Mexican-Hat, Morlet, Coiflet1 y Symlet2
FUENTE: Extraído de [Badrzadeh \(2014\)](#)

2.3.5. Análisis multiresolución

Para aplicar la transformada *wavelet* a una serie de datos numéricos, se hace necesario implementar una TWD. La idea fue desarrollada por Mallat (1989), y se basa en el análisis multiresolución. Mallat diseñó un algoritmo basado en un banco de filtros que permite obtener una transformada *wavelet* en forma rápida a partir de los datos de interés (Acevedo 2009). Según la teoría de Mallat, la serie temporal discreta x_t se descompone en una serie de señales de Aproximación y Detalle (Wu *et al.* 2009; Santos *et al.* 2014).

a. Filtros de un nivel

En la mayoría de las señales son las componentes de baja frecuencia las que le otorgan a la señal la mayor parte de su información, o bien, le dan una especie de identidad a la señal. Mientras que las componentes de alta frecuencia se encargan de incorporar características más particulares. Es por ello que se subdividen las componentes de una señal en dos categorías:

- Aproximaciones (baja frecuencia)
- Detalles (alta frecuencia)

Luego, surge la idea de separar estas dos componentes a través de filtros. Lo anterior queda ejemplificado en el diagrama de la Figura 25.

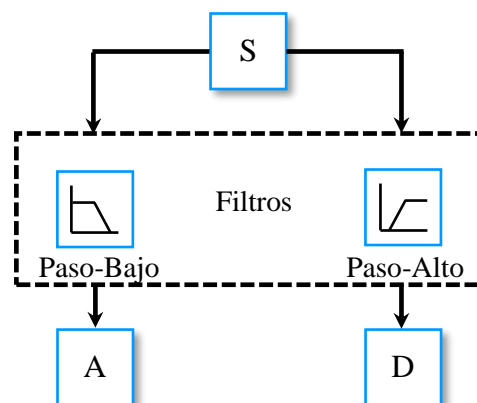


Figura 25: Diagrama de descomposición de señales usando bancos de filtros

FUENTE: Adaptado de Acevedo (2009)

Donde S es la señal que se desea analizar, A la salida del paso-bajo y D la salida del filtro paso-alto. Naturalmente, los filtros son diseñados de tal manera que sean complementarios, es decir, la suma de A y D debe ser S . Si se diseñaran los filtros en forma muy separada se

perdería información, o en caso contrario se estaría amplificando la banda de entrecruzamiento.

Sin embargo, este procedimiento tiene la desventaja que se aumenta al doble el número de datos originales, pues por cada muestra de S se genera un par de muestras (A, D), por lo que el costo matemático y computacional se incrementa. Para remediar esta dificultad se propone un método que guarda la mitad de los puntos (A, D), sin perder en ello información de la señal S . Este procedimiento es conocido como submuestreo o decimación. La idea se ilustra en la Figura 26.

Los círculos con flechas representan la eliminación de datos o submuestreo. Luego, cD y cA son los nuevos coeficientes obtenidos de la etapa de filtración. Intuitivamente se puede concluir que al tener cD y cA , en conjunto, se tiene la misma cantidad de datos que las de la señal original S , y se ha mantenido la información necesaria. En la Figura 26 se ejemplifica la idea para una señal S de 1000 datos, obteniéndose en la salida dos series de aproximadamente 500 datos cada una.

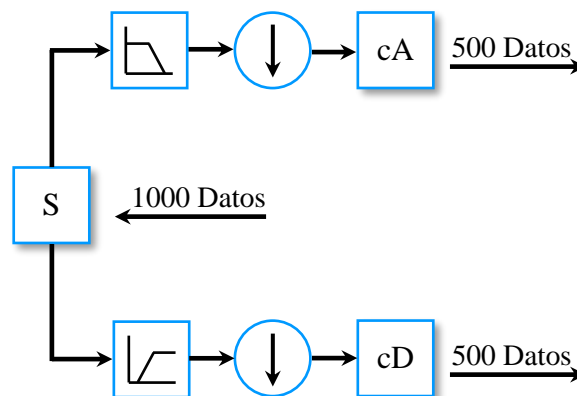


Figura 26: Descomposición de señales usando bancos de filtros y submuestreo
FUENTE: Adaptado de [Acevedo \(2009\)](#)

Para muchas señales de mayor complejidad, no basta con dos bandas de frecuencias (alta y baja), sino que más bien debe hacerse una descomposición de más niveles para poder separar las características y poder analizarlas independientemente. Surge la idea entonces de filtros multiniveles.

b. Filtros multinivel

Para realizar la motivación expuesta en el punto anterior, basta con iterar el proceso de filtrado, es decir, aplicar el mismo procedimiento a las señales de salida de la primera

etapa, y así sucesivamente hasta el nivel de precisión que se desee. Lo anterior da origen a una descomposición multinivel conocida como ramificación o árbol de descomposición *wavelet*, cuya idea es expuesta en la Figura 27. Note que cD_1 resulta ser la componente de más alta frecuencia de la señal, y cA_3 la de menor frecuencia. Al ser descompuesta la señal en mayor cantidad de bandas de frecuencia se posee una información más detallada acerca de S , por lo que esta metodología es conocida como multiresolución. Surge en forma inmediata la inquietud acerca del diseño del algoritmo, relativo al número de niveles a utilizar (Acevedo 2009).

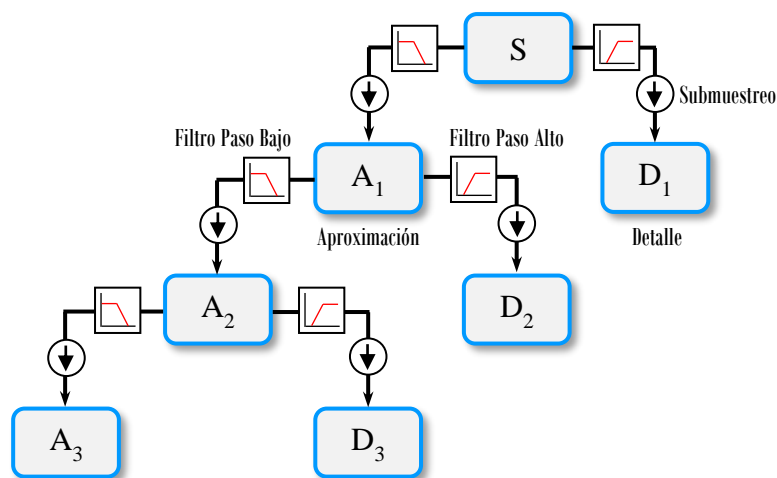


Figura 27: Árbol de descomposición *wavelet* (descomposición multinivel)
FUENTE: Adaptado de Santos *et al.* (2014)

2.3.6. Determinación del número de niveles

Podría pensarse en forma intuitiva que se obtienen resultados óptimos con un mayor número de niveles de descomposición, pero, esto no siempre es así. En (Coifman y Wickerhauser 1992) se recomienda una ramificación que vaya de acuerdo a la naturaleza de la señal a estudiar, o bien elegir métodos que buscan la descomposición óptima, como, por ejemplo, el de la entropía (Santos y Silva 2013).

2.4. ENFOQUES DE MODELAMIENTO HÍBRIDO

El uso de técnicas híbridas se ha extendido recientemente a predicciones de series de tiempo con buenos resultados (Abreu 2015). Los hallazgos tanto teóricos como empíricos en la literatura muestran que la combinación de diferentes métodos puede ser una manera efectiva y eficiente de mejorar los pronósticos (Aladag *et al.* 2009). Por lo tanto, el uso de

modelos híbridos o la combinación de varios modelos se ha convertido en una práctica común para superar las limitaciones de los modelos individuales y mejorar la precisión del pronóstico ([Khashei et al. 2012](#)).

2.4.1. Introducción a las técnicas híbridas

Tradicionalmente, el modelo autorregresivo integrado de media móvil o ARIMA (acrónimo del inglés *autoregressive integrated moving average*) ha sido uno de los modelos lineales más utilizados en la predicción de series de tiempo. Sin embargo, el modelo ARIMA no puede capturar fácilmente los patrones no lineales ([Pai y Lin 2005](#)). Pueden ser inapropiados si el mecanismo subyacente no es lineal. De hecho, los sistemas del mundo real son a menudo no lineales. En los últimos 20 años, las redes neuronales se introdujeron como herramientas eficientes para el pronóstico de series temporales. Las RNA son no lineales y pueden aproximar cualquier función continua, utilizado cada vez más para la predicción y pronóstico en recursos hídricos ([Yan y Zou 2013](#)).

Sin embargo, la mayoría de los estudios fueron aplicaciones simples de utilizar enfoques tradicionales de series temporales y RNA. Muchas de las series temporales de la vida real son extremadamente complejas para ser modeladas usando enfoques simples, especialmente cuando se requiere alta precisión ([Yan y Zou 2013](#)). Los hallazgos tanto teóricos como empíricos han sugerido que la combinación de diferentes modelos puede ser una manera efectiva de mejorar el performance predictivo de cada modelo individual, especialmente cuando los modelos en el conjunto son bastante diferentes ([Zhang 2007](#); [Khashei et al. 2009](#)).

La literatura sobre este tema se ha expandido drásticamente desde los primeros trabajos de [Bates y Granger \(1969\)](#) y [Clemen \(1989\)](#). [Wedding y Cios \(1996\)](#) describieron una metodología de combinación utilizando redes de función de base radial y modelos ARIMA de Box-Jenkins. [Voort et al. \(1996\)](#) introdujeron un método híbrido llamado KARIMA utilizando un mapa auto-organizado de *Kohonen* (SOM) y el método ARIMA para la predicción a corto plazo. [Van den Boogaard et al. \(1998\)](#) propusieron un modelo híbrido que incluía una RNA y un modelo ARMAX y lo aplicaron para predecir la profundidad de un lago. [Medeiros y Veiga \(2000\)](#) consideraron un sistema híbrido de pronóstico de series temporales con RNA usadas para controlar los parámetros variables en el tiempo de un modelo autorregresivo de transición suave.

Abrahart y See (2000) presentaron una solución de red híbrida basada en la agrupación de los registros hidrológicos con una red neuronal de SOM. Pai y Lin (2005) propusieron una metodología híbrida para explotar la fuerza única de los modelos ARIMA y máquinas de vectores de soporte para la previsión de precios de las acciones. Khashei *et al.* (2008), basado en los conceptos básicos de las RNA, propusieron un nuevo modelo híbrido de las RNA para superar la limitación de datos y producir un modelo de pronóstico más preciso, especialmente en situaciones de datos incompletos. Khashei *et al.* (2009) presentaron un enfoque híbrido ARIMA y la inteligencia artificial para la predicción de los mercados financieros.

Recientemente, muchos estudios proporcionan buenas descripciones de los modelos híbridos ARIMA-RNA. Zhang (2003) propuso una metodología híbrida ARIMA y RNA para aprovechar las dos técnicas lineal y no lineal y aplicó el modelo híbrido propuesto a tres tipos de series temporales, y demostró que el modelo combinado puede ser una manera efectiva de mejorar los pronósticos logrados por cualquiera de los modelos utilizados por separado. Khashei y Bijari (2010) propusieron un nuevo modelo híbrido y dieron resultados más precisos que el modelo híbrido de Zhang y también los modelos ARIMA y RNA utilizados separadamente en dos horizontes temporales diferentes. Faruk (2010) utilizó el modelo híbrido propuesto por Zhang para hacer la predicción de la calidad del agua y proporcionó una precisión mucho mejor sobre los modelos ARIMA y RNA.

Ambos modelos ARIMA y RNA han logrado éxitos en sus propios dominios lineales o no lineales, respectivamente. Sin embargo, ninguno de ellos es un modelo universal que es adecuado para todas las circunstancias. La aproximación de los modelos ARIMA a problemas no lineales complejos puede no ser adecuada. Por otro lado, el uso de RNA para modelar problemas lineales ha dado resultados mixtos. Por lo tanto, no es aconsejable aplicar RNA ciegamente a cualquier tipo de datos. Dado que es difícil conocer completamente las características de los datos en un problema real, la metodología híbrida que tiene capacidad de modelar series de tiempo tanto lineales como no lineales puede ser una buena estrategia para el uso práctico. Mediante la combinación de diferentes modelos, diferentes aspectos de los patrones subyacentes pueden ser capturados (Zhang 2003).

2.4.2. Enfoques de la inteligencia artificial

Para mejorar la precisión y confiabilidad de la predicción de series hidrológicas, se ha estudiado un gran número de técnicas de modelamiento, que van desde el enfoque “basados

en la física” hasta el enfoque “basados en datos”. Como resultado de los avances tecnológicos en los últimos años, una gran cantidad de investigación se ha llevado a cabo en las áreas que se pueden clasificar en términos generales como la inteligencia artificial (IA). Recientemente, los enfoques de la IA se han vuelto cada vez más populares en el modelamiento y pronóstico de series de tiempo hidrológico no lineal. Hay varias definiciones de la IA; Sin embargo, la definición adoptada por [Kingston et al. \(2008\)](#), afirman que la IA es el "estudio de las formas en que las computadoras pueden realizar tareas que demandan comportamiento inteligente"; por ejemplo, aprendizaje, razonamiento, toma de decisiones y resolución de problemas.

Se sabe que las técnicas de la IA tienen grandes habilidades en la estimación de series de tiempo no lineales en las que han atraído la atención de diversas áreas, incluida la ingeniería hidrológica. En el campo hidrológico, emplean series temporales históricas disponibles para simular el sistema. En este aspecto, la RNA y algunos otros métodos han sido la principal atención para modelar sistemas hidrológicos no lineales. Independientemente de su éxito relativo, los procedimientos de RNA tienen algunas carencias que limitan sus aplicaciones en muchos problemas incluyendo las estimaciones de caudales máximos. Cuando las fluctuaciones de la señal no son estacionarias, los enfoques basados en RNA no son satisfactorios en precisión y en esta situación el performance del modelo carece de la calidad esperada. Con el fin de superar esta deficiencia y aumentar la precisión de las estimaciones del modelo, se sugieren algunos modelos híbridos en combinación con RNA ([Nourani et al. 2011a](#); [Darlane et al. 2014](#)).

El último par de décadas han visto un progreso notable en la capacidad de desarrollar modelos hidrológicos precisos. Entre los diversos modelos conceptuales y de caja negra desarrollados durante este período, los modelos híbridos *wavelet* y la IA se encuentran entre los más prometedores en la simulación de procesos hidrológicos. [Badrzadeh \(2014\)](#), desarrolló modelos híbridos de la IA, aplicando el análisis multiresolución *wavelet* en combinación con las técnicas de la IA, a datos históricos de cuatro ríos australianos, con características diferentes. Los resultados confirmaron la robustez de los modelos híbridos son muy superiores a los modelos clásicos de la IA en la predicción de eventos extremos repentinos. Esto se puede atribuir a la utilidad de las transformadas *wavelets* en el análisis multiresolución, eliminación de ruido y detección de efectos de borde sobre una señal, así como la fuerte capacidad de los métodos de la IA en la optimización y predicción de procesos ([Nourani et al. 2009b](#); [Nourani et al. 2014a](#)). Asimismo, en las tareas de

pronóstico, el método híbrido *wavelet* y la inteligencia artificial sigue un procedimiento de dos pasos (Figura 28):

- Uso de la transformada *wavelet* para preprocesar datos de entrada. Esto incluye proporcionar una representación tiempo-frecuencia de una señal en diferentes períodos en el dominio del tiempo, así como información considerable sobre la estructura física de los datos.
- Extracción de características de la señal principal para servir como entradas de la IA, y permitir que el modelo completo procese los datos.

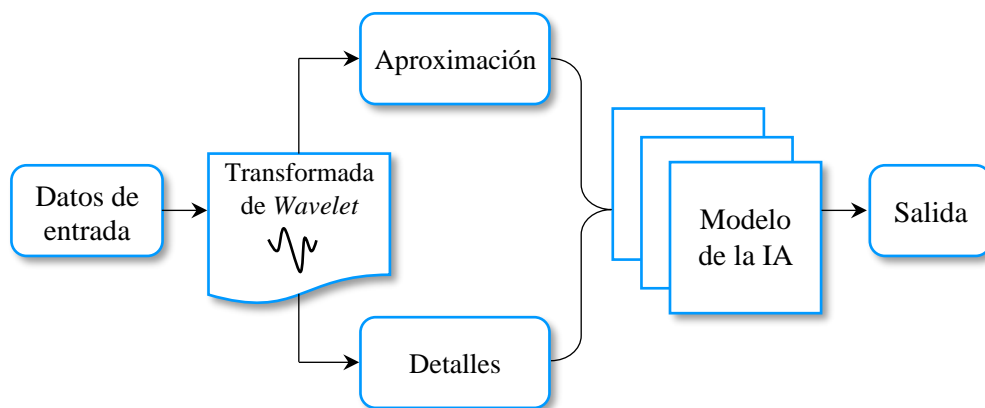


Figura 28: Esquema del modelo de pronóstico híbrido *wavelet* - IA
FUENTE: Extraído de [Nourani et al. \(2014a\)](#)

Los métodos “basados en datos” de la IA se muestran prometedores en el modelamiento y pronóstico de procesos hidrológicos no lineales y en el manejo de grandes cantidades de conjuntos de datos dinámicos y ruidos ocultos. Tales propiedades de los modelos basados en la IA son adecuadas para problemas de modelamiento hidrológico. Se han utilizado numerosas herramientas o técnicas de la IA, incluyendo versiones de optimización de búsqueda, optimización matemática, así como métodos basados en lógica, clasificación, aprendizaje estadístico y probabilidad. En particular, tres subconjuntos de la IA han sido ampliamente utilizados en los campos hidroclimatológicos y ambientales:

- **Computación evolutiva:** una rama de métodos de optimización que incluye algoritmos de la inteligencia de enjambre tales como optimización de colonia de hormiga (ACO) u optimización de enjambre de partículas (PSO) y algoritmos evolutivos tales como algoritmos genéticos (GA), programación de expresión genética (GEP) y programación genética (GP).

- **Lógica difusa:** los sistemas difusos (*Fuzzy systems*) pueden utilizarse para un razonamiento incierto, que proporcionan una perspectiva lógica en las técnicas de la inteligencia artificial.
- **Clasificadores y métodos de aprendizaje estadístico:** estos modelos emplean enfoques estadísticos y de aprendizaje automático. Los clasificadores más utilizados son las RNA, métodos del *Kernel* tales como la máquina de vector de soporte (SVM), algoritmos de k-vecinos más cercanos como el mapa autoorganizado (SOM), modelo de mezcla *Gaussiana*, clasificador ingenuo de *Bayes* y árbol de decisión. Las RNA y el método de la IA, se usan en hidrología a través de dos enfoques: i) supervisados, incluyendo RNA acíclicos o *feedforward* (donde la señal pasa en una sola dirección) y RNA recurrentes (que permiten retroalimentación), y ii) sin supervisión (p. ej., SOM).

Entre las aplicaciones más amplias de los métodos de la IA; GA, GP, *Fuzzy*, RNA y SVM son ampliamente utilizados en diferentes campos de la hidrología. Desde su aparición en la hidrología, el desempeño eficiente de las técnicas de la IA, tales como modelos “basados en datos”, ha sido reportado sobre un amplio rango de procesos hidrológicos (p. ej., precipitación, caudal, precipitación-escorrentía, carga de sedimentos, agua subterránea, sequía, fusión de nieve, evapotranspiración, calidad de agua, etc.). El número de investigadores activos en esta área ha aumentado significativamente en la última década, así como el número de publicaciones (Nourani *et al.* 2014a). Una revisión exhaustiva de las aplicaciones de técnicas de la IA en hidrología se puede encontrar en Kingston *et al.* (2008); Alagha *et al.* (2012); Nourani *et al.* (2014a) y Badrzadeh (2014).

2.4.3. Aplicaciones de modelos híbridos *wavelet* - IA

A pesar del buen desempeño de los modelos “basados en datos” (*data-driven* en inglés) en la simulación de relaciones hidrológicas no lineales, estos modelos pueden no ser capaces de hacer frente a los datos no estacionarios si no se realiza el procesamiento previo de los datos de entrada y/o salida (Shoib *et al.* 2016). Por otro lado, se ha encontrado que la aplicación de la transformada de *wavelet* (TW) es eficaz en el tratamiento de datos no estacionarios y se ha convertido en una técnica de análisis popular para mejorar el desempeño de los modelos de RNA debido a su capacidad para revelar simultáneamente información espectral y temporal dentro de una señal (Nourani *et al.* 2009a).

La TW es una herramienta matemática que considera simultáneamente la información espectral y temporal contenida en los datos. La TW descompone los datos de la serie temporal principal en sus subcomponentes y estos subconstituyentes se utilizan como entradas externas a los modelos “basados en datos”. El modelo resultante es conocido como el modelo híbrido *wavelet*-“basados en datos”. Estos modelos híbridos, que utilizan datos de entrada a escala múltiple, dan como resultado un mejor desempeño capturando información útil oculta en los datos de la serie temporal principal en su forma cruda. Una revisión exhaustiva de las aplicaciones de TW en hidrología se puede encontrar en [Nourani et al. \(2014a\)](#).

El modelo híbrido *wavelet* redes neuronales (WRN) fue propuesto por primera vez por [Aussem et al. \(1998\)](#) para predecir una serie temporal financiera. En hidrología y gestión de recursos hídricos, [Kim y Valdés \(2003\)](#) aplicaron transformadas de *wavelet* diádica y RNA para predecir sequías en la cuenca del río Conchos en México y mostraron que el modelo de conjunción mejoró significativamente la capacidad de las RNA para pronosticar la sequía regional. [Wang y Ding \(2003\)](#) aplicaron el modelo WRN para predecir los niveles de agua subterránea poco profunda en Beijing y la descarga diaria del río Yangtze en China, con resultados que sugirieron que el modelo podría aumentar la precisión del pronóstico y extender el tiempo de predicción.

[Partal y Cigizoglu \(2008\)](#) estimaron el sedimento suspendido diario usando un modelo WRN. En un paso, datos de sedimentos anteriores y en el otro paso, se usaron datos de caudal de río diarios para estimar la carga de sedimento. Informaron que el modelo de WRN tiene un buen ajuste a los datos observados (especialmente en valores pico) y tiene mejores resultados que el modelo de RNA clásico. [Kisi \(2008\)](#) aplicó un modelo de *neuro-wavelet* para pronosticar (un mes de anticipación) los caudales de los ríos Canakdere y Goksudere en Turquía y comparó los resultados con aquellos modelos de *perceptrón* multicapa (MLP), regresión lineal múltiple (MLR) y autoregresión (AR) solos, revelando que el modelo combinado de WRN podría aumentar tanto la precisión y el desempeño del pronóstico.

Los métodos de WRN también se han utilizado en el monitoreo de la calidad del agua ([Kim et al. 2006](#)), precipitación ([Partal y Kisi 2007](#)), modelamiento mensual de fluctuaciones de nivel ([Kisi 2009](#)), y la evapotranspiración ([Partal 2009](#)). [Adamowski y Sun \(2010\)](#) también aplicaron un método acoplada de transformada *wavelet* y RNA para la predicción de flujo en una cuenca semiárida. Utilizaron los coeficientes *wavelet* como

entradas en los modelos de RNA para pronósticos de 1 y 3 días. Sus resultados mostraron que los modelos *wavelet*-RNA proporcionan pronósticos de flujo más precisos que los modelos de RNA. Krishna *et al.* (2011) propusieron un modelo híbrido denominado modelo WRN, que combinaron el análisis *wavelet* y RNA, para el modelamiento de series de tiempo de caudales del río Malaprabha, India. Compararon con modelos clásicos de RNA y autoregresivos (AR) y concluyeron que los pronósticos del modelo híbrido WRN fueron más precisos que los modelos RNA y AR.

Kisi (2011) investigó la habilidad técnica del modelo híbrido WRN para modelar el caudal mensual de las estaciones Gerdelli e Isakoy en la región oriental del Mar Negro de Turquía. Los modelos WRN fueron desarrollados mediante la combinación de dos métodos, red neuronal de regresión generalizada (GRNN) y transformada de *wavelet* discreta. Los resultados de la comparación revelaron que el WRN tiene mejores resultados que los modelos GRNN y red neuronal *feedforward* en la predicción mensual de caudal. Wei *et al.* (2012) desarrollaron un método de modelamiento híbrido WRN para el pronóstico de caudal del río Weihe en China, usando datos mensuales de series temporales. Mostraron que los modelos WRN desempeñaron un pronóstico más precisa en comparación a los modelos RNA solos.

La ventaja de la técnica *wavelet* es que proporciona un proceso matemático para descomponer una señal en múltiples niveles de detalles y se puede hacer un análisis de estos detalles. El análisis *wavelet* puede diagnosticar efectivamente las señales del componente de frecuencia principal y la información local abstracta de la serie temporal. En la última década, se introdujo la teoría *wavelet* en el análisis del procesamiento de señales. En los últimos años, las transformaciones *wavelet* se han aplicado con éxito al análisis de datos de olas y otras aplicaciones de ingeniería oceánica. Además, se introdujo la teoría *wavelet* en el campo de la hidrología. El análisis *wavelet* se identificó recientemente como una herramienta útil para describir las series temporales de lluvia y escorrentía (Krishna *et al.* 2011).

La combinación de transformada *wavelet* y RNA es uno de los métodos híbridos más comunes mencionados en la literatura. La estructura de WRN consiste en la transformada *wavelet* y el modelo RNA como se explicó en párrafos anteriores. Los modelos híbridos WRN son los modelos de RNA donde se utilizan las subseñales descompuestas como entradas. En esta estructura, el proceso de propagación posterior se utiliza para calcular los pesos mediante el entrenamiento del modelo híbrido WRN. Estas subseñales descompuestas

se obtienen por transformada *wavelet* discreta debido a un menor esfuerzo computacional que la transformación *wavelet* continua (Darlane *et al.* 2014).

2.4.4. Directrices generales para los modelos híbridos

Shoaib *et al.* (2016), señalaron que el desempeño de los modelos híbridos *wavelet* “basados en datos” es muy dependiente de la selección de numerosos factores. Estos factores incluyen la selección del tipo de transformada de *wavelet* - TW (continua o discreta). Igualmente, la TW requiere una función *wavelet* madre para realizar la transformación. El desempeño de los modelos híbridos es también extremadamente sensible a la selección de la función de *wavelet* apropiada de una serie de funciones de *wavelet* disponibles. La eficiencia del desempeño de los modelos híbridos *wavelet* depende en gran medida de la selección del nivel de descomposición adecuado. Además, la elección del modelo “basados en datos” es otro factor vital ya que existen diferentes tipos de estos modelos disponibles, incluyendo la red neuronal de *perceptrones* multicapa. Por lo tanto, en las siguientes secciones se formula las directrices generales para la implementación satisfactoria de modelos híbridos *wavelet*-“basados en datos”.

a. Selección del tipo de transformada de *wavelet*

Como se dijo anteriormente, existen dos tipos de transformadas de *wavelet*, llamados, la TWC y la TWD. Diferentes estudios han utilizado ambos, como fueron mencionados en Shoaib *et al.* (2016). Utilizaron la TWC con la función *wavelet Morlet* para identificar la variabilidad temporal de los datos de lluvia-escorrentía. Aplicaron la TWC con la *wavelet Morlet* y la TWD con la *wavelet Haar* para explicar la no estacionariedad de los datos de una cuenca kárstica. Usaron la TWD con el algoritmo ‘á trous’ para hacer predicciones en hidrología. Desarrollaron modelos de predicción de caudales a un día de anticipación y modelos acoplados *wavelet* para la previsión de caudales diarios utilizando TWC.

Sin embargo, la aplicación de TWC se asocia con el demérito hereditario de generación de gran cantidad de datos debido al cálculo de coeficientes de TWC en cada escala y traslación. Además, también se argumentó a favor de TWD que el hidrólogo no tiene a su disposición un proceso de señal de tiempo continuo, sino más bien una señal de tiempo discreto. Por lo tanto, la mayoría de los modelos acoplados *wavelets* se limitan al uso de TWD. Shoaib *et al.* (2016) indicaron que aplicaron TWD para la predicción de precipitación. Aplicaron TWD para la predicción y pronóstico de caudales. Desarrollaron

modelos híbridos *wavelet*-RNA utilizando tanto TWC y TWD con la función *wavelet* db4. Compararon el desempeño de la TWC y TWD para los modelos de precipitación-escorrentía acoplados a *wavelets*. Finalmente, abogaron el uso de TWD sobre TWC debido a su performance superior.

b. Selección de la función *wavelet*

El desempeño de los modelos híbridos WRN también depende de la selección de un tipo particular de funciones *wavelet* madre. Como se dijo anteriormente, existen diferentes familias de *wavelets* disponibles y cada uno contiene un número de funciones *wavelet* madre. En general, un número limitado de estudios evaluó los efectos de los tipos de función *wavelet* sobre el desempeño de los modelos de RNA. [Nourani et al. \(2009a\)](#) encontraron que las *wavelets* db4 y *Meyer* proporcionan un desempeño superior a que de la *wavelet Haar* al desarrollar un modelo *wavelet*-RNA para la predicción de precipitación mensual. Similarmente, [Nourani et al. \(2011a\)](#) desarrollaron modelos de precipitación-escorrentía multivariado de *wavelet*-RNA y probaron el desempeño de siete funciones *wavelet*, llamados funciones *Haar*, db2, db3, db4, *Sym2*, *Sym3* y *Coif1*. Los resultados mostraron que las funciones *wavelet* madre *Haar* y db2 proporcionan mejores resultados que el resto de las cinco funciones *wavelet* aplicadas.

En el contexto de modelos de pronóstico de inundación *wavelet*-RNA, [Singh \(2011\)](#) encontró que la función *wavelet* db2 produce mejor desempeño que la función db1. De manera similar, [Maheswaran y Khosa \(2012\)](#) observaron que la función db2 tiene mejor desempeño que las cinco funciones *wavelet*, llamados, funciones db1, db3, db4, *Sym4* y B-spline al desarrollar modelos de predicción hidrológica *wavelet-Volterra*. El desempeño de 23 funciones *wavelet* madre fueron probados en un estudio realizado por [Shoaib et al. \(2014a\)](#). Se encontró que la función *wavelet* db8 de la familia *wavelet Daubechies* superaba a las otras funciones *wavelet* probadas en el estudio.

c. Selección del nivel de descomposición

La elección del nivel/etapa de descomposición (L) adecuado en el desarrollo de modelos “basados en datos” acoplados a *wavelet* es muy vital ya que se relaciona con las características estacionales y periódicas incorporadas en los datos hidrológicos. El número máximo de niveles de descomposición depende de la longitud de datos disponibles. Una descomposición de TWD consiste de Log_2N niveles a lo más.

Aussem *et al.* (1998), Wang y Ding (2003) y Nourani *et al.* (2009a, b) utilizaron la siguiente fórmula dada en la Ecuación (2.27) para calcular el L adecuado.

$$L = \text{ent}[\log(N)] \quad (2.27)$$

donde L es el nivel de descomposición, N es el número de datos de la serie temporal y ent significa número entero. Esta ecuación se derivó para los datos totalmente autorregresivos, considerando sólo la longitud de los datos sin prestar atención a la firma estacional del proceso hidrológico Nourani *et al.* (2011a).

Asimismo, algunos otros estudios hidrológicos previos utilizaron diferentes niveles de descomposición sin dar ningún razonamiento lógico. Adamowski y Sun (2010) utilizaron una descomposición de nivel ocho para desarrollar modelos de redes neuronales acopladas a *wavelet*, Partal y Kisi (2007) utilizaron descomposición en diez niveles para desarrollar un modelo *Neuro-Fuzzy* acoplado a *wavelets*. Sin embargo, Kisi y Shiri (2011) utilizaron tres niveles de descomposición para desarrollar modelos *wavelet* basada en programación genética y *Neuro-Fuzzy*.

Shoaib *et al.* (2014a, b; 2015) sugirieron el uso de nueve niveles de descomposición de datos para el desarrollo de modelos híbridos *wavelet*-“basados en datos”. La descomposición en nueve niveles contiene una subseñal de aproximación (a9) a gran escala (frecuencia más baja) y nueve subseñales de detalles (d1, d2, d3, d4, d5, d6, d7, d8 y d9) a pequeña escala (frecuencia más alta). La subserie de detalle d4 corresponde al modo de 16 días de la serie de tiempo, d5 al modo de 32 días (cerca de modo mensual), d6 al modo de 64 días, d7 al modo de 128 días (cerca de cuatro meses), d8 al modo de 256 días (cerca de ocho meses y medio) y d9 al modo de 512 días (unos diecisiete meses). Además, la descomposición en nueve niveles contiene las subseries d8 y d9 que son responsables de detectar variaciones estacionales en los datos de lluvia de entrada casi de forma anual. Esta periodicidad anual se considera el ciclo estacional extremadamente significativo y principal en los datos de series de tiempo hidrológicas.

d. Selección del modelo “basados en datos”

Hay varios tipos de modelos “basados en datos” disponibles y la selección de un tipo adecuado es una de las tareas más importantes y difíciles en el desarrollo de modelos híbridos *wavelet*. Los modelos red neural de *perceptrón* multicapa (MLP), red neural de *feedforward* generalizada (GFF), red neural de función base radial (RBF), red neural

modular (MNN), sistema de inferencia adaptativa *neuro-fuzzy* (ANFIS) y programación de expresión génica (GEP) son los más comúnmente utilizados en hidrología.

Diferentes estudios emplearon diferentes tipos de redes neuronales para mapear procesos hidrológicos no lineales. El tipo MLP se ha utilizado en muchos estudios de precipitación-escorrentía, igualmente emplearon el tipo RBF para modelar procesos de precipitación-escorrentía. El potencial de ANFIS ha sido explorado en muchos estudios hidrológicos (Shoaib *et al.* 2014b). La red neural modular es otro tipo de red neural que se ha utilizado con éxito para el modelamiento de la precipitación-escorrentía. En los últimos años, el enfoque “basados en datos” conocido como la programación genética (GP), basado en la computación evolutiva surgió como una poderosa herramienta de modelado para resolver problemas hidrológicos y de recursos hídricos. La GEP es una variante de GP y se ha encontrado que muestra un mejor desempeño que otros enfoques “basados en datos” como la RNA y el ANFIS (Shoaib *et al.* 2016).

Durante la última década, varios estudios aplicaron con éxito TW con el fin de aumentar la eficiencia de pronóstico de los modelos de red neuronal. Sin embargo, se encuentra que el tipo MLP es el tipo de red neural más ampliamente utilizado para el propósito de desarrollar modelos de red neuronal acoplados a *wavelet*.

e. Identificación de subseries *wavelet* dominantes

Una deficiencia importante asociada con el modelo acoplado a *wavelet* es el uso de un gran número de entradas, ya que TW descompone los datos de entrada en varias series de subdatos. El uso de grandes cantidades de entradas no sólo aumenta el tiempo de simulación, sino que también aumenta la complejidad computacional y dificulta la calibración de la red. Además, algunas de las series de datos obtenidas por la TW contienen más información sobre la firma hidrológica de la cuenca, mientras que otras pueden ser simplemente ruidosas o contener información no/menos significativa. La identificación de subseries de datos *wavelets* que contienen información significativa sobre el sistema es también otro tema importante en la implementación exitosa de modelos hidrológicos acoplados a *wavelet* (Shoaib *et al.* 2016).

El análisis del coeficiente de correlación lineal entre la subserie de entrada/predictor y la salida ha sido ampliamente utilizado para la determinación de importantes subseries de *wavelet* que se utiliza como entrada para el desarrollo de modelos de RNA acoplados a

wavelet. Además, se encuentran dos enfoques diferentes en la literatura que emplean coeficientes de correlación lineal para este propósito.

- En el primer enfoque, se realiza un análisis de correlación entre cada subserie descompuesta y la salida observada. Las subseries *wavelet* que tienen una fuerte correlación con la salida observada solamente se consideran como la entrada para los modelos de RNA acoplados a *wavelet*, mientras que las que tienen una correlación débil simplemente se ignoran. Este enfoque se ha utilizado con éxito en algunos de los estudios hidrológicos.
- Del mismo modo, otro enfoque con respecto a la selección de subseries *wavelet* dominantes basado en el análisis de correlación cruzada también se ha empleado en el desarrollo de modelos hidrológicos RNA y *Neuro-Fuzzy*. En este enfoque, se realiza un análisis de correlación entre cada subserie descompuesta y la salida observada. Las subseries de *wavelets* que tienen una correlación muy débil se ignoran y la nueva serie de datos de entrada se obtiene sumando las subseries *wavelets* que tienen buena correlación con la salida observada. Esta nueva serie de datos se utiliza posteriormente como entrada para el desarrollo de modelos de RNA acoplados a *wavelets*.

Sin embargo, [Nourani et al. \(2011b, 2013\)](#) critican la selección de subseries *wavelets* dominantes sobre la base de la correlación lineal, ya que puede existir una fuerte relación no lineal entre la entrada y salida objetivo a pesar de la presencia de una correlación lineal débil. Además, algunos otros estudios probaron diferentes métodos matemáticos para seleccionar subseries *wavelet* dominantes en el desarrollo de modelos acoplados a *wavelet* en hidrología. [Nourani et al. \(2014b\)](#) abogó por el uso de mapas de características autoorganizados (SOP) y enfoques basados en entropía para seleccionar la subserie *wavelet* dominantes para desarrollar un modelo de lluvia-escorrentamiento acoplado a *wavelet*.

Es, por lo tanto, obvio de la literatura anterior citada, que algunos de los estudios hidrológicos anteriores utilizaron todas las subseries *wavelet* como entrada, mientras que otros identificaron las subseries *wavelets* dominantes usando diferentes métodos y posteriormente los usaron como entrada en el desarrollo de modelos acoplados a *wavelet*. En consecuencia, es vital comparar estas diferentes estrategias para el desarrollo de modelos basados en *wavelets*, de modo que se pueda identificar una estrategia óptima ([Shoab et al. 2016](#)).

III. MATERIALES Y MÉTODOS

En este Capítulo, se describe la aplicación de dos técnicas de modelamiento, tanto la red neuronal tradicional (RNA) como el híbrido *wavelet* red neuronal (WRN) para el pronóstico de caudales diarios a corto plazo en una parte (estación hidrométrica de Tamshiyacu) de la cuenca del río Amazonas, situada a 40 km aguas arriba de la ciudad de Iquitos, en Perú. Esto implica abordar las principales áreas que incluyen preprocesamiento de datos, elección de adecuadas entradas para el modelo, elección de una apropiada geometría de red, estimación de parámetros y validación del modelo. Sin embargo, en la mayoría de las aplicaciones de redes neuronales en la hidrología, el proceso de construcción del modelo se ha descrito mal o llevado a cabo de forma incorrecta (Maier y Dandy 2000). El propósito de este Capítulo III fue ilustrar el proceso de construcción y/o desarrollo de los modelos de RNA y modelos híbridos WRN.

3.1. DESCRIPCIÓN DEL ÁMBITO DE ESTUDIO

La cuenca del Amazonas, la más importante de los ríos del Planeta en términos de superficie drenada y producción, cubre 6 112 000 km² (cerca del 5 por ciento de las tierras continentales) y vierte al Atlántico un volumen de agua que representa alrededor un 15 por ciento de las contribuciones hídricas a los océanos (Molinier *et al.* 1996). Asimismo, esta cuenca es una de las regiones con mayor precipitación y fuente de humedad en el mundo, sobre ello es posible tener eventos hidrológicos extremos. A su vez, la variabilidad interanual de los caudales en el curso principal del Amazonas puede causar inundaciones o niveles hídricos muy bajos (Espinoza *et al.* 2009b).

Además, esta cuenca, situada entre 5° de latitud norte y 20° de latitud sur, se extiende sobre siete países: Brasil (63 por ciento), Perú (16 por ciento), Bolivia (12 por ciento), Colombia (5.6 por ciento), Ecuador (2.3 por ciento), Venezuela (0.6 por ciento) y Guyana (0.2 por ciento). Esta limitado al norte por los relieves del escudo de Guyana, al oeste por la Cordillera de Los Andes, al sur por la meseta del escudo brasileño, y al este por el Océano Atlántico. Entre Los Andes y los escudos antiguos, los límites de la cuenca no

son muy marcadas y se observan fenómenos de difluencia: en el norte con el Orinoco y en el sur hacia el río de La Plata (Molinier *et al.* 1996).

El Perú es un país esencialmente amazónico debido a que más del 76 por ciento (977 900 km²) de su territorio pertenece a la cuenca del Amazonas y el 98 por ciento de los recursos hídricos del país se encuentran en la amazonía (DGAS 1995). La presente investigación se enfoca sobre la amazonía del Perú, en consecuencia, se seleccionó la estación hidrométrica Tamshiyacu (Figura 29, derecha), debido a su ubicación estratégica, a su importancia para la caracterización de los eventos hidrológicos extremos y a la información histórica más larga.

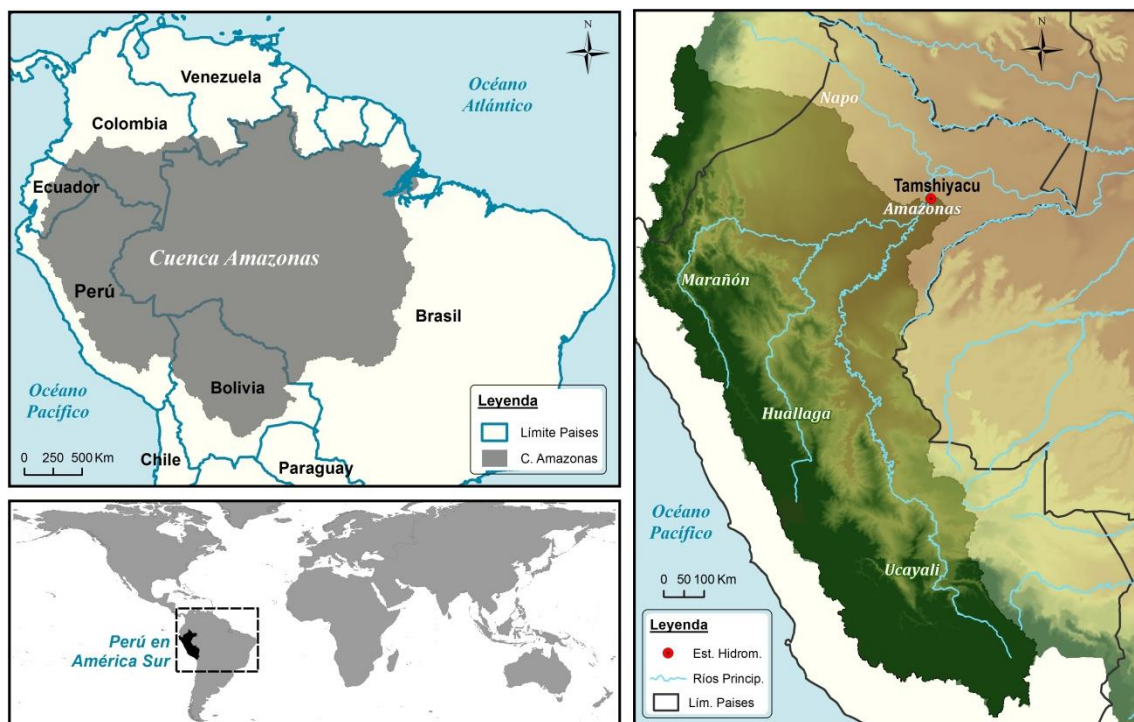


Figura 29: Ubicación de Perú en América del sur (izq. inferior), cuenca del Amazonas en América del sur y países involucrados (izq. superior), estación hidrométrica de Tamshiyacu dentro de la amazonía peruana (derecha)

3.1.1. ESTACIÓN HIDROMÉTRICA DE TAMSHIYACU

La estación de Tamshiyacu (4.00° sur y 73.16° oeste) es la primera estación de aforo en la corriente principal del río Amazonas, situado aguas abajo de la confluencia de los ríos Marañón y Ucayali, a 50 km aguas arriba de la ciudad de Iquitos; drena un área de cuenca de 750 000 km² (53 por ciento en Los Andes) y tiene un caudal medio anual de 32 000 m³/s (Espinoza *et al.* 2006; Espinoza *et al.* 2011b; Guyot *et al.* 2007). Además, es el principal tributario occidental de la cuenca del Amazonas, presentando la serie histórica

de niveles más larga, con datos desde 1984 (Espinoza *et al.* 2006, 2011b). También, su ubicación es estratégica, lo que permite la caracterización de un período de sequías o inundaciones en la población amazónica.

La serie de caudales diarios en Tamshiyacu muestra que el caudal más bajo fue observado en setiembre de 2010 (8 300 m³/s), una rápida transición hacia uno de los caudales más altos fue observado en mayo 2011 (49 500 m³/s). Finalmente, en abril del 2012, durante el siguiente periodo de aguas altas, el río Amazonas experimentó su caudal histórico más elevado (55 400 m³/s) (Espinoza *et al.* 2012).

3.2. INFORMACIÓN DISPONIBLE Y RECURSOS

La serie hidrológica analizada consiste en una colección de 27 años de información diaria de caudales, que fue proporcionado por el Servicio Nacional de Meteorología e Hidrología (SENAMHI) del Perú, como parte del observatorio ORE-HYBAM (SENAMHI-IRD-IGP-UNALM, www.ore-hybam.org). La serie corresponden a los caudales medios diarios registrados en la estación hidrométrica Tamshiyacu (río Amazonas) en el periodo comprendido entre el 1 de setiembre de 1984 al 18 de febrero del 2013. Sin embargo, para la presente investigación se consideró el periodo comprendido entre el 1 de setiembre de 1985 al 31 de agosto del 2012, siendo un total de 9 862 muestras diarias.

Por otro lado, todos los análisis y evaluaciones se realizaron a través de programas (*scripts* en inglés) asociados que están escritos, principalmente, en lenguaje de programación *R* (v3.1.3) y *Matlab* R2017a (abreviatura de *MATrix LABoratory*). Además, se ha utilizado el *RStudio* (v0.98.1091) que es un entorno de desarrollo integrado (IDE) que hace más fácil el trabajo con *R*. También, fue extensamente utilizado el *package R* llamado *hydroTSM*, que proporciona funciones para la gestión, el análisis, la interpolación y el trazado de las series de tiempo utilizados en hidrología y relacionados a ciencias ambientales. En particular, este paquete está altamente orientado a tareas de modelamiento hidrológico. Asimismo, fue utilizado extensamente el *Toolbox Wavelet*TM y *Neural Network*TM en *Matlab* (R2017a), que permitió desarrollar los preprocesamientos de datos, construir, estimar parámetros y validar los modelos de RNA y modelos híbridos WRN.

3.3. ANÁLISIS PRELIMINAR DE DATOS

El análisis exploratorio de datos (*EDA* por sus siglas en inglés) es a menudo el primer paso en un análisis estadístico formal, ya que ayuda a entender las principales características de la muestra particular que un analista está usando (Tukey 1977). En ocasiones, las descripciones o resúmenes inteligentes de los datos pueden ser suficientes para cumplir los propósitos para los que se recopilaron los datos. Los resúmenes efectivos también pueden apuntar a datos "malos" o aspectos inesperados que podrían pasar desapercibidos si los datos son tratados ciegamente por las computadoras.

El *EDA* se desarrolló a finales de 1970 cuando los gráficos por computadora se convirtieron en ampliamente disponibles. Puesto que, el *EDA* es un enfoque/filosofía para el análisis de datos que emplea una variedad de técnicas (principalmente gráficos) para:

- Maximizar la comprensión de un conjunto de datos;
- Descubrir la estructura subyacente;
- Extraer variables y/o patrones importantes;
- Detectar valores atípicos (*outliers*) y anomalías;
- Probar los supuestos subyacentes;
- Desarrollar modelos parsimoniosos; y
- Determinar la configuración óptima de los factores.

Tukey (1977) indica que la gran ventaja del *EDA* es que permite mirar y pensar en los datos, y que puede revelar patrones que las pruebas estadísticas estándar pasarían por alto debido a su énfasis en modelos específicos. Sin embargo, el *EDA* no es una fórmula mágica para interpretar datos sin el riesgo de eludir información relevante. Sólo el sentido común y la precaución pueden mantenerlo en la zona entre ignorar patrones interesantes y sobreinterpretarlos.

3.4. DESARROLLO DE MODELOS DE REDES NEURONALES ARTIFICIALES

Enfoques de modelamiento de RNA han sido acogidos con entusiasmo por los profesionales en hidrología y modelamiento de recursos hídricos, ya que se perciben superar algunas de las dificultades asociadas con ciertos enfoques estadísticos tradicionales. La aplicación efectiva de las RNA requiere una apreciación de los méritos relativos de las diferentes redes disponibles, así como una comprensión de las mejores maneras de entrenarlos. Tipos de red y algoritmos de entrenamiento están en constante evolución y el neurohidrólogo debe

mantenerse actualizado. En general, la aplicación involucra en primer lugar, la separación de los datos disponibles en dos subseries de entrenamiento y de validación. Posteriormente a través de un proceso iterativo de entrenamiento de RNA, se determina la mejor configuración de la red (número de nodos de entrada, número de capas ocultas y número de nodos por capa oculta).

En esta investigación se estructuró solamente modelos univariados de RNA, debido a la limitada disponibilidad de datos diarios extensos de variables hidrológicas en esta región amazónica. En consecuencia, se consideró sólo la dependencia del valor actual y sus rezagos del caudal diario como la única variable hidrológica. Es decir, se usaron rezagos como predictores (entradas a la red, iniciándose de lo más básico hasta lo más complejo) para pronosticar un valor futuro (salida objetivo, que representa a cada uno de los múltiples horizontes de pronóstico) de la misma variable.

3.4.1. Protocolo para la implementación de las RNA

Es evidente que no existe un marco riguroso para la aplicación de las RNA para el modelamiento hidrológico. La Figura 30 proporciona los principales pasos en el proceso de modelamiento de las RNA. En cada etapa, hay varias alternativas disponibles para los modeladores, que ofrece una gran flexibilidad, pero también puede crear dificultades ya que no hay directrices claras para indicar bajo qué circunstancias se deben adoptar enfoques particulares. Otra dificultad es el hecho de que la lista de alternativas de modelamiento está creciendo rápidamente, ya que continuamente se proponen nuevas arquitecturas de red y algoritmos de optimización en la literatura de RNA.

Para complicar aún más las cosas, existen interdependencias entre las distintas etapas en el proceso de modelamiento. Por ejemplo, el algoritmo de optimización tiene un impacto en la velocidad de entrenamiento y la precisión de la predicción, la arquitectura de red tiene un impacto en la velocidad de procesamiento y la cantidad de datos necesarios para estimar los pesos de conexión de manera eficiente, el criterio de parada y el método utilizado para determinar el número óptimo de nodos de la capa oculta afecta la forma en que se debe dividir la serie temporal y la función de transferencia en la capa de salida influye en el rango al que se deben escalar los datos (Maier y Dandy 2000; Govindaraju y Rao 2000; Dawson y Wilby 2001).

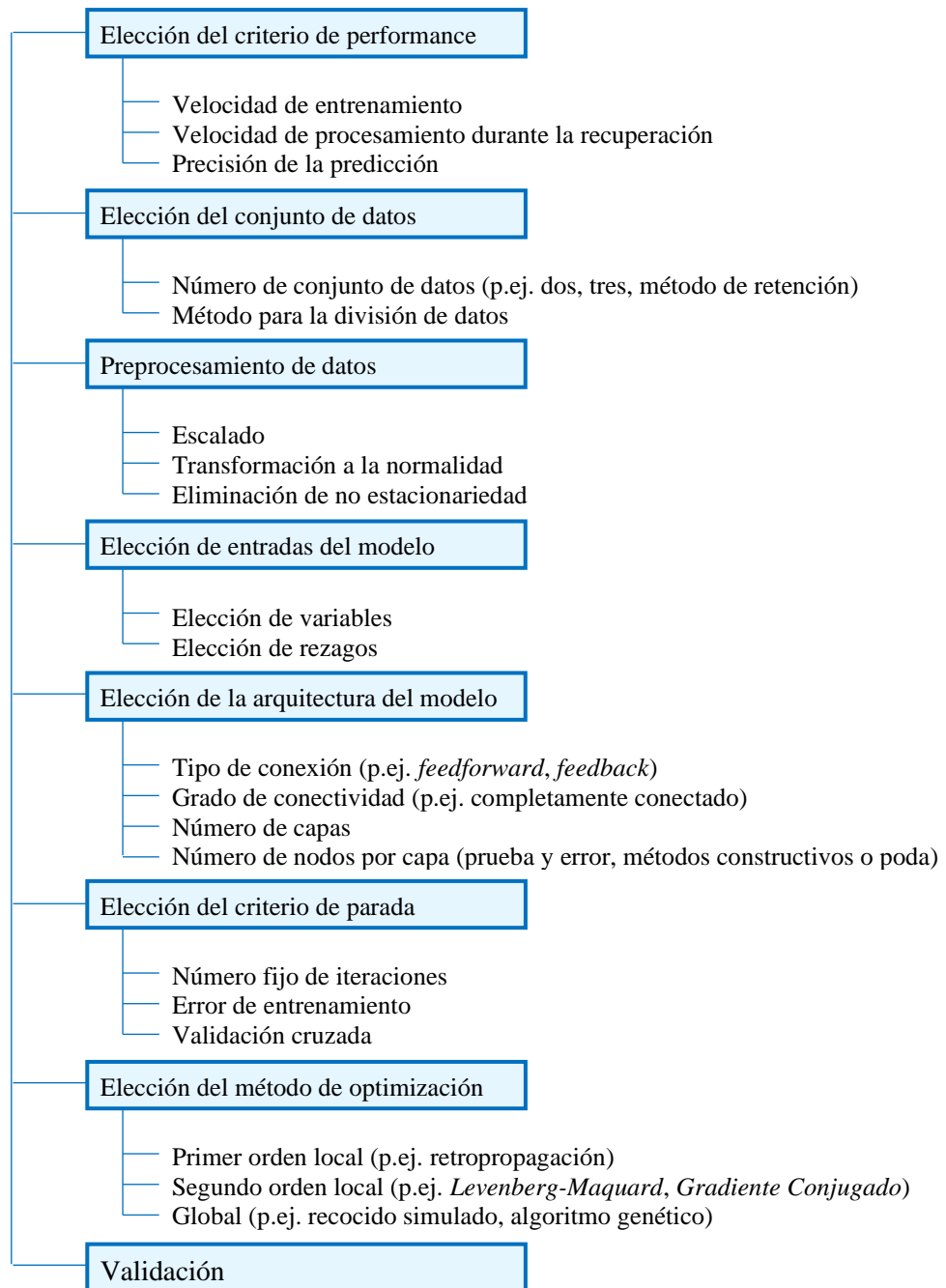


Figura 30: Principales pasos en el desarrollo de modelos de RNA
FUENTE: Traducido de [Maier y Dandy \(2000\)](#)

El enfoque de la Figura 30, se centra en el preprocesamiento de datos, la determinación de las entradas del modelo y la elección de la arquitectura del modelo.

El proceso de construcción del modelo se ve, por lo tanto, principalmente influenciado por los siguientes aspectos:

- La selección del algoritmo de aprendizaje, entrenamiento u optimización utilizado y de los valores de sus parámetros, los cuales, impactan directamente aspectos, como el tiempo de convergencia y la robustez del modelo final.
- La selección del número adecuado de neuronas en la capa oculta.
- La selección de las entradas relevantes.
- Y la selección de cuál de los posibles modelos es el más adecuado.

Se sabe que estos aspectos están interrelacionados, y que influyen en la bondad del modelo para ajustarse a los datos, así como en su capacidad de generalización. La multiplicidad de criterios en cada una de las fases del proceso de construcción del modelo hace que la técnica sea inestable, ya que cualquier cambio en un solo criterio conduce a grandes cambios en el ajuste (Yu *et al.* 2009). Consecuentemente, un análisis del comportamiento de estos factores es de especial interés a la hora de implementar una RNA, muchos de los cuales requieren del juicio experto del pronosticador.

A continuación, se ofrece una breve explicación y justificación de aplicación/uso de cada una de estas etapas y las adoptadas en esta investigación.

a. División de la data

Es una práctica común dividir los datos disponibles en dos subseries; una serie de entrenamiento y una serie de validación independiente. Por lo general, las RNA son capaces de extrapolar más allá del rango de los datos utilizados para el entrenamiento. En consecuencia, se puede esperar pronósticos deficientes cuando los datos de validación contienen valores fuera del rango de los utilizados para el entrenamiento. También es imperativo que las series de entrenamiento y validación sean representativas de la misma población (Maier y Dandy 2000).

Esta tipología de criterios se basa a menudo en dividir la muestra de datos en al menos dos series principales: ajuste o entrenamiento, y prueba o pronóstico. La serie de entrenamiento se usa para estimar los parámetros adecuados del modelo, mientras que la serie de pronóstico para verificar la validez y utilidad del modelo. Dentro de este enfoque usualmente el 80 por ciento de los datos se usan para la calibración de los modelos, mientras que el 20 por ciento restante de los datos son utilizados para la prueba y validación de los modelos, con 10 por ciento usado para la prueba y 10 por ciento usado para la validación (Maier y Dandy 2000; Bowden *et al.* 2002; Balayneh y Adamowski

2012). Sin embargo, existen diferentes métodos de división de datos que tienen efecto directo sobre el desempeño de las RNA (Bowden 2003).

En esta investigación, se empleó una técnica convencional de división de datos, mediante la cual la serie de datos se subdividió de forma arbitraria y no se consideraron las propiedades estadísticas de los respectivos subseries de datos. Este enfoque es consistente con los utilizados en muchos artículos sobre la aplicación de las RNA a las variables de recursos hídricos.

b. Arquitectura del tipo *perceptrón* multicapa

En la fase de diseño de la arquitectura, también denominada diseño de la topología, se determina el número de neuronas que tendrá la red, así como su disposición en capas y la conectividad entre las mismas. A partir de un análisis de los datos del problema, se determinan cuántas entradas y salidas tiene la red, así como el número de neuronas y cómo estas están distribuidas en capas e interconectadas entre sí.

La topología de la red debe adecuarse al problema a resolver, y la no existencia de técnicas que realicen esta función hace que haya que recurrir a la experiencia y a la técnica de ensayo y error, probando varias topologías distintas, para finalmente conseguir una que se adapte de forma satisfactoria al problema (Gestal 2013).

La arquitectura de la red neuronal más popular empleada en el modelamiento y pronóstico hidrológico es del tipo *perceptrón* multicapa (MLP), debido a su simplicidad (Maier y Dandy 2000), el cual ha sido adoptado en esta investigación. Este tipo de redes neuronales no cuenta con ningún tipo de retroalimentación por lo que la salida de la red es calculada directamente desde el vector de entradas mediante la modificación de los pesos sinápticos de las conexiones hasta encontrar una solución óptima.

En la Figura 31, se presenta el esquema representativo de una red *perceptrón* multicapa. Este tipo de arquitectura de red neuronal consta de:

- Una capa de entrada, donde existe una neurona por cada valor rezagado de la serie de tiempo y_k , con un número P de retardos (es decir: y_{k-1}, \dots, y_{k-P}). Transmitida a la capa oculta mediante las conexiones $\alpha_{p,h}$.
- Una o más capas ocultas. En el caso particular analizado se consideró únicamente una capa oculta con H neuronas, justificado su uso por Kisi (2004) y Wu *et al.* (2009). Cada neurona de la capa oculta transforma su entrada neta usando una función no lineal

$g(\cdot)$, conocida técnicamente como función de activación o de transferencia, y la envía hacia la siguiente capa a través de las conexiones β_h .

- Una capa de salida. Para este caso particular la capa de salida consta de una sola neurona; la salida de la red corresponde al valor actual de la serie de tiempo y_k .

Tanto las capas ocultas como la capa de salida reciben un peso unitario de una neurona que no recibe las entradas, la cual es notada en la Figura 31 con la letra B . Las conexiones por las que se transmite este pulso son notadas como ω_h para la capa oculta y como η para la neurona de la capa de salida.

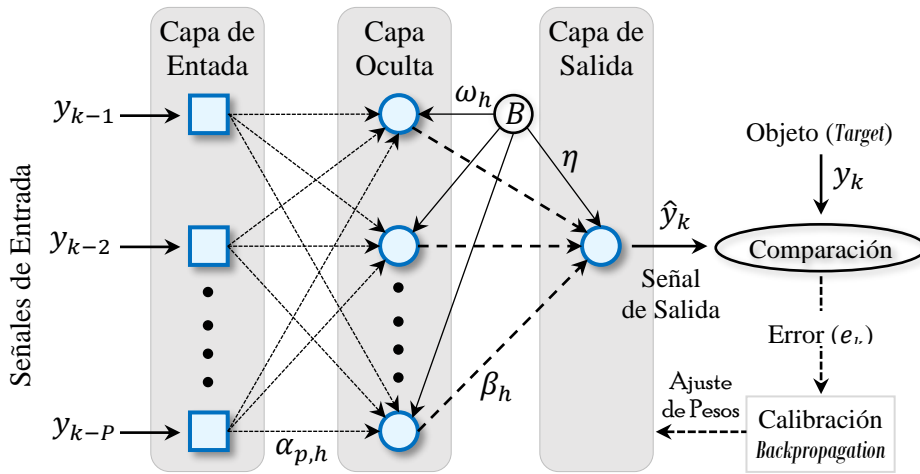


Figura 31: Esquema de un *perceptrón* multicapa con una capa oculta
FUENTE: Adaptado de [Sánchez \(2012\)](#)

Así, el valor actual de una serie de tiempo \hat{y}_k es una función no lineal de sus valores pasados y_{k-1}, \dots, y_{k-p} , la cual se define como:

$$\hat{y}_k = \eta + \sum_{h=1}^H \left\{ \beta_h \times g \left[\omega_h + \sum_{p=1}^P \alpha_{p,h} y_{k-p} \right] \right\} + e_k \quad (3.1)$$

donde e_k representa los errores o residuos del modelo que son independientes e idénticamente distribuidos (iid) con media cero y varianza constante σ_e^2 .

El problema de pronóstico es formulado como una tarea en la cual las unidades de entrada están compuestas por valores rezagados de la serie temporal (predictores individuales), y la salida es la predicción del valor siguiente en la secuencia temporal (pronóstico a un paso adelante), de tal forma que el comportamiento del sistema se rige por la ecuación:

$$y_{k+1} = f(y_k, y_{k-1}, \dots, y_{k-p}) \quad (3.2)$$

Para representar una serie de tiempo mediante redes *perceptrón* multicapa, se establece el valor de la serie en el instante de tiempo k y sus P valores rezagados como entradas a la red en el mismo tiempo k . Se establece, a su vez, el valor de salida de la red como el valor de la serie de tiempo en el instante $k+1$. De este modo, el entrenamiento de la red neuronal minimiza una medida de error entre la salida de la red, para cualquier instante k , y el valor de la serie en el instante $k+1$. Una vez modelada la serie de tiempo mediante la red, para realizar el pronóstico de los valores futuros a un paso adelante bastará con utilizar las propias salidas de la red como entradas en el tiempo siguiente, de modo que se sigan cumpliendo las ecuaciones iniciales supuestas:

$$\begin{aligned} y_{k+1} &= f(y_k, y_{k-1}, \dots, y_{k-P}) \\ y_{k+2} &= f(y_{k+1}, y_k, \dots, y_{k-P+1}) \\ &\vdots \end{aligned} \quad (3.3)$$

También, esta etapa alberga el determinar las funciones de activación y/o transferencia. No obstante, hay algunas reglas heurísticas para la selección de la función de activación. Por ejemplo, utilizar funciones de activación logística para problemas de clasificación que implican el aprendizaje sobre el comportamiento de la media, y utilizar funciones de tangente hiperbólica es la opción más popular si el problema envuelve el aprendizaje acerca de las desviaciones de la media, tales como el problema de pronóstico. Sin embargo, no está claro si las diferentes funciones de activación tienen efectos importantes en el desempeño de las redes (Zhang *et al.* 1998). Así pues, en esta investigación se ha utilizado la función tangente sigmooidal hiperbólica para la capa oculta y la función de activación lineal para la capa de salida.

c. Preprocesamiento de datos

A fin de acelerar los procesos de aprendizaje de la red, es recomendable que a los datos de entrada se les realice un preprocesamiento o normalización. Las siguientes formulas son generalmente aplicadas en la normalización de los datos de entrada:

- Transformación lineal entre [0, 1]:

$$x_n = \frac{(x_0 - x_{\min})}{(x_{\max} - x_{\min})} \quad (3.4)$$

- Transformación lineal entre [a, b]:

$$x_n = (b - a) \frac{(x_0 - x_{\min})}{(x_{\max} - x_{\min})} + a \quad (3.5)$$

- Normalización estadística:

$$x_n = \frac{(x_0 - \bar{x})}{s} \quad (3.6)$$

- Normalización simple:

$$x_n = \frac{x_0}{x_{m\acute{a}x}} \quad (3.7)$$

donde x_0 y x_n representan la data sin procesar y la normalizada, $x_{m\acute{i}n}$, $x_{m\acute{a}x}$, \bar{x} y s representan el mınimo, maximo, promedio y la desviacion estandar de los datos, respectivamente.

El proceso de normalizacion de los datos de salida generalmente es independiente al de los datos de entrada, aunque para el pronostico de series temporales se recomienda que la normalizacion sea de forma conjunta entre los datos de entrada y los datos objetivo. Se debe tener en cuenta en el proceso de normalizacion, los rangos de salidas de las funciones de activacion de los nodos. El rango de salida de la funcion logıstica es de [0, 1], el de la tangente hiperbolica es de [-1, 1]. Varios estudios utilizan rangos de normalizacion que varan de [0.1, 0.9] o [0.2, 0.8] para funciones logısticas, basado en que las funciones de activacion tienen lımites asintoticos y para no limitar la capacidad de extrapolacion de la red (Zhang *et al.* 1998). En esta investigacion, se ha normalizado en el rango de [-1, 1] debido al empleo de la funcion de activacion tangente hiperbolica.

Se debe considerar que cuando se utilizan datos normalizados, los valores de salida de la red se encuentran escalados en el rango de normalizacion, por lo que para evaluar la bondad del modelo se deben reescalar los datos (Zhang *et al.* 1998).

d. Seleccion de entradas relevantes

La seleccion de las variables de entrada depende en gran medida del conocimiento que posea el modelador acerca de la serie de tiempo, y es tarea de este elegir segun algun criterio previamente fijado la necesidad de cada variable dentro del modelo. Aunque, no existe una vıa sistematica para determinar el conjunto de entradas aceptada por la comunidad investigativa, recientes estudios han planteado la utilizacion de procedimientos racionales, basados en el uso de analisis decisional, o en los tradicionales metodos estadısticos, tales como las funciones de autocorrelacion (Tang y Fishwick 1993).

Sin embargo, resultados mixtos acerca de los beneficios de incluir muchas o pocas variables de entrada son también reportados en la literatura. [Zhang et al. \(2001\)](#) afirman que el número de variables de entrada en un modelo de RNA para pronóstico es mucho más importante que el número de neuronas ocultas. Otras técnicas han sido también propuestas basadas en análisis heurístico de la importancia de cada rezago, pruebas estadísticas de la dependencia no lineal, criterios de identificación del modelo, etc.

Las metodologías expuestas para la selección de entradas relevantes se basan en el supuesto que el algoritmo de entrenamiento está adecuadamente desarrollado, lo que degenera en consecuencias relacionadas con el incumplimiento de este supuesto, donde no se puede garantizar que la selección de las variables sea correcta y más aún que no se incluyan variables irrelevantes ([Sánchez 2012](#)).

Un método comúnmente empleado de determinación es utilizar un conocimiento *a priori* del sistema bajo investigación para seleccionar un conjunto de entradas de modelo candidato. Sin embargo, cuando el número potencial de variables de entrada a una RNA es grande y no hay conocimiento *a priori* disponible para sugerir qué variables incluir, el proceso de selección es inherentemente difícil. En muchas aplicaciones de RNA, los profesionales prestan muy poca atención a la tarea de la selección de entrada ([Maier y Dandy 2000](#); [Bowden 2003](#)).

En esta investigación, se utilizó el análisis de correlación simple para determinar la fuerza relacional correspondiente entre las series de tiempo de entrada (distintos rezagos) y las series de tiempo de salida (diferentes horizontes de pronóstico).

e. Selección de neuronas de la capa oculta

El proceso de selección del número adecuado de neuronas ocultas se ha convertido en un importante tópico de investigación en redes neuronales toda vez que estas pueden conducir a resultados ampliamente diferentes. Un número pequeño de neuronas en la capa oculta hace que la red no puede aprender adecuadamente las relaciones existentes en los datos, mientras que un gran número hace que la red memorice los datos con una pobre generalización y una poca utilidad para el pronóstico. Algunos autores proponen que el número de neuronas ocultas debe ir en función del número de entradas relevantes, pero, este criterio está a su vez relacionado con la extensión de la serie temporal y de los conjuntos de entrenamiento, validación y pronóstico ([Sánchez 2012](#)).

Tradicionalmente, el número de nodos ocultos para los modelos de RNA se selecciona mediante un método de prueba y error. Sin embargo, en un estudio de [Wanas *et al.* \(1998\)](#) determinaron empíricamente que el mejor desempeño de una RNA se produce cuando el número de nodos ocultos es igual a $\log(T)$, donde T es el número de muestras de entrenamiento. Otro estudio realizado por [Mishra y Desai \(2006\)](#) determinaron que el número óptimo de nodos ocultos es $2n+1$, donde n es el número de nodos de entrada.

En esta investigación, inicialmente el número de neuronas ocultas se estimó tomando como referencia los resultados de los dos métodos descritos anteriormente, posteriormente, se tomaron valores cercanos al valor de referencia, inferiores y superiores. En conclusión, se seleccionó el número óptimo por ensayos de prueba y error.

f. Criterios de parada de entrenamiento

Los criterios utilizados para decidir cuándo detener el proceso de entrenamiento son de vital importancia, ya que determinan si el modelo ha sido entrenado óptimamente o subóptimamente. Ejemplos de este último incluye detener el entrenamiento demasiado temprano o ha ocurrido un sobreajuste de los datos de entrenamiento ([Maier y Dandy 1998](#)). En esta etapa, es importante recordar que el sobreajuste está íntimamente ligado a la razón entre el número de muestras de entrenamiento con el número de los pesos de conexión. [Amari *et al.* \(1997\)](#) muestran que el sobreajuste no se produce si la relación anterior es superior a 30. En tales casos, el entrenamiento puede ser detenido cuando el error de entrenamiento ha alcanzado un valor suficientemente pequeño o cuando los cambios en el error de entrenamiento siguen siendo pequeños. Cuando no se cumple la condición anterior, hay claros beneficios en el uso de validación cruzada ([Amari *et al.* 1997](#)).

En términos prácticos, esto no es una tarea fácil, y no ha habido mucha discusión sobre los méritos relativos de la utilización de la validación cruzada como un criterio de parada. Por ejemplo, existen estudios que discuten los beneficios y la efectividad de la utilización de métodos de validación cruzada. Sin embargo, algunos investigadores sugieren que es imposible determinar el óptimo tiempo de parada y que existe el peligro de que el entrenamiento sea detenido prematuramente, para mayor detalle revisar [Maier y Dandy \(2000\)](#).

g. Entrenamiento de la red neuronal

El objetivo de una red neuronal es “generalizar” una función de la forma:

$$Y^n = f(X^m, p) \quad (3.8)$$

donde Y^n , es el vector n dimensional de salidas y X^m , es el vector m dimensional de entradas, p representa los parámetros de la red.

El proceso de entrenamiento consiste en un problema de minimización no lineal de la función de error, a partir de la modificación de los pesos de la red. La existencia de diversos métodos de optimización tiene como consecuencia la existencia de diversos algoritmos de entrenamiento. Sin embargo, a la fecha no existe ningún método que garantice un óptimo global ya que todos los métodos de optimización tienen la posibilidad de quedar atrapados en mínimos locales.

El método de aprendizaje más popular en redes de varias capas se conoce como retropropagación (*backpropagation*), fue desarrollado por Bryson y Ho en 1969, y desarrollado también en la tesis doctoral no publicada de Werbos (1974). Este método permaneció ignorado hasta mediados de los años 80, saliendo a la luz con el trabajo de Rumelhart *et al.* (1986) denominado “*Learning internal representation by error propagation*”. El método está basado en la minimización de la función de error, mediante el ajuste de los pesos de la red. Se utiliza la técnica del descenso del gradiente, en el cual se determina la respuesta de la red a los datos de entrada para los patrones de entrenamiento, estos valores se comparan con los datos deseados determinándose el error mediante la evaluación de la función de error seleccionada, la cual generalmente es la suma cuadrática del error, luego el error es propagado hacia atrás a través de cada nodo de la red, mediante la siguiente ecuación:

$$\Delta w_{ij}(n) = -\varepsilon \cdot \frac{\partial E}{\partial w_{ij}} + \alpha \cdot \Delta w_{ij}(n-1) \quad (3.9)$$

donde $\Delta w_{ij}(n)$ y $\Delta w_{ij}(n-1)$ es el ajuste realizado al peso de la conexión entre los nodos i y j , para la época n y $n-1$ respectivamente. ε y α son los denominados *ratio* y *momentum* de entrenamiento.

El *momentum* puede acelerar el entrenamiento en zonas donde la superficie de error es plana y ayuda en evitar oscilaciones en los pesos. Un valor pequeño de la *ratio* de entrenamiento supone una aproximación lenta hacia el mínimo, mientras que un valor

grande acelera la búsqueda del mínimo; sin embargo, puede generar problemas de aproximaciones. La *ratio* de entrenamiento es importante en evitar que el proceso de entrenamiento se ubique en un mínimo local.

El método de retropropagación tiene dos fases, la primera consistente en la determinación de la respuesta de la red a partir de los datos de entrada y la posterior evaluación del error. La segunda fase consiste en propagar el error hacia los nodos de inicio modificando los pesos de las conexiones utilizando la ecuación 3.9.

En esta investigación, para el entrenamiento de las RNA *feedforward* (es decir, MLP de tamaño moderado) se empleó el algoritmo de *Levenberg-Marquardt* (LM), debido a sus ventajas de una convergencia más rápida en comparación con el tradicional algoritmo de descenso de gradiente ampliamente usado (Wei *et al.* 2012). Aunque este requiere una mayor cantidad de memoria que otros algoritmos (Adamowski y Sun 2010), es menos probable de ser atrapado en los problemas de mínimos locales (Badrzadeh *et al.* 2013).

h. Optimización del MLP

Entrenar un MLP se caracteriza por ser de aprendizaje supervisado, usualmente es un gran problema numérico de optimización, en el cual se desean estimar los parámetros o pesos óptimos de la red (Masters 1993). Los términos entrenamiento, aprendizaje u optimización de una red neuronal hace alusión a la misma idea, es decir, la de modificar los pesos del MLP de manera que coincida la salida deseada por el usuario con la salida obtenida por la red ante la presentación de un determinado patrón o estímulo de entrada, la cual se mide a través de una función de error de ajuste. En otras palabras, el entrenamiento de un MLP es un problema de minimización no lineal en el cual los pesos de la red son iterativamente modificados con el fin de minimizar el error entre la salida deseada y la obtenida.

En la práctica, se requiere de un procedimiento iterativo para resolver el problema constructivo y entrenamiento de la red neuronal, donde el valor de los pesos w (parámetros), el número de entradas P y de neuronas ocultas H , sean calculados con miras a minimizar el error de ajuste. Wu *et al.* (2009) señala que el propósito del entrenamiento de la red es optimizar los pesos w y los sesgos h .

Se asume como punto inicial una red con topología mínima y se calculan los pesos óptimos de tal modelo; posteriormente, se adicionan parámetros al modelo (incrementando entradas o neuronas ocultas) y se optimizan los pesos del nuevo modelo resultante; este proceso se

repite hasta obtener una solución aceptable. El problema de optimización de los pesos, como se dijo anteriormente, es independiente del método constructivo, por lo que puede ser cualquiera de los algoritmos de entrenamiento mencionados en la literatura.

i. Validación o prueba de los modelos

Una vez completado la fase de entrenamiento viene la fase de validación, durante la cual la red deberá ser capaz de reconocer o evaluar y dar respuestas correctas ante patrones de estímulos diferentes a los presentados durante la fase de entrenamiento, sólo que ahora no se realizará ningún ajuste en los pesos sinápticos. En otras palabras, una vez terminado el aprendizaje, una red puede generalizar; es decir, ante entradas similares a las de su entrenamiento, producirá salidas correctas.

Si la diferencia en el error obtenido usando la subserie de validación es marcadamente diferente a la obtenida usando la subserie de datos de entrenamiento, es probable que las dos subseries de datos no sean representativos de la misma población o que el modelo haya sido sobreajustado. La validación deficiente también puede deberse a la arquitectura de la red, una falta de, o inadecuada, preprocesamiento de datos y normalización de datos de entrenamiento/validación (Maier y Dandy 2000).

Gestal (2013) señala que, para conseguir una mejor capacidad de generalización, los datos usados para el entrenamiento deben cubrir un rango de hechos suficientemente amplio. En general, cuando aumenta el tamaño y variedad de hechos de optimización disminuye la necesidad de que los datos de entrada durante la fase de validación se parezcan mucho a los patrones de entrenamiento; es decir, la red generalizará mejor.

3.4.2. Medidas estadísticas de desempeño

Una medida de precisión, se define a menudo en términos del error de pronóstico (ε_t), que es la diferencia entre el caudal observado (Q_{obs}) y el caudal pronosticado (Q_{sim}), definido mediante la ecuación 3.10:

$$\varepsilon_t = Q_{obs,t} - Q_{sim,t}, \quad t = 1, 2, \dots, T \quad (3.10)$$

En consecuencia, una medida apropiada de precisión para un problema dado no es universalmente aceptada por los académicos y profesionales de pronóstico. Hay una serie de medidas de precisión en la literatura de pronóstico y cada una tiene ventajas y limitaciones (Zhang *et al.* 1998).

Sin embargo, Dawson *et al.* (2007) discutieron 20 medidas de desempeño generalmente utilizados en el pronóstico hidrológico. En esta investigación, principalmente se seleccionó y empleó cuatro índices estadísticos los más frecuentemente utilizados para evaluar el desempeño de los modelos de pronóstico:

- Raíz del error cuadrático medio (RMSE),

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{n}} \quad (3.11)$$

- Error absoluto medio (MAE),

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t| \quad (3.12)$$

- Error absoluto relativo medio (MARE) y

$$MARE = \frac{1}{n} \sum_{t=1}^n \frac{|y_t - \hat{y}_t|}{y_t} \quad (3.13)$$

- Eficiencia de Nash-Sutcliffe (NSE).

$$NSE = 1 - \frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{\sum_{t=1}^n (y_t - \bar{y}_t)^2} \quad (3.14)$$

donde y_t e \hat{y}_t son los valores observados y pronosticados, \bar{y}_t es el valor medio de la data observada y_t , $t = 1, 2, 3, \dots, n$. Se obtienen valores más pequeños de error, cuanto más cerca están los valores pronosticados a los valores observados.

La RMSE proporciona una buena medida de bondad de ajuste en caudales altos, mientras que el MAE mide una perspectiva más equilibrada de la bondad de ajuste en caudales moderados. Los modelos con bajo RMSE son tratados como los mejores modelos (Dadu y Deka 2013).

El MARE se basa en un principio similar de MAE, pero relativo a los datos observados, que muestra un comportamiento coherente, robusto, descriptivo (interpretación clara) y lógico, y por lo tanto puede utilizarse como un indicador ideal para la evaluación de modelos bajo diversas condiciones de salida. En general, para el valor de MARE en por ciento, las siguientes calificaciones se pueden usar como guía: 0-5 es excelente, 5-10 es

muy bueno, 10-15 es bueno, 15-20 es razonable, 20-25 es moderado y >25 es insatisfactorio (Ali y Abustan 2014).

Una mejora sobre el índice estadístico “coeficiente de determinación” o R^2 y que se ha utilizado muy comúnmente en hidrología es la eficiencia de *Nash-Sutcliffe* (NSE), introducida por Nash y Sutcliffe (1970). Así es, un valor de NSE de 0.9 y por encima es muy satisfactorio, de 0.8 a 0.9 representa un modelo bastante bueno, y por debajo de 0.8 se considera insatisfactorio (Dawson y Wilby 2001).

El desempeño de la red se evalúa comparando los valores de la salida deseada y los generados por la red tras el proceso de entrenamiento, la cual se mide mediante el error de ajuste. Se han definido índices del error, los cuales permiten definir criterios para la finalización del proceso de entrenamiento y realizar comparaciones entre distintos modelos propuestos.

3.4.3. Implementación de las RNA en Matlab

Las generalidades del *software Matlab*, se encuentran en el Anexo 1. Además, para trabajar con redes neuronales, seguramente se puede encontrar con una simple búsqueda en Internet un gran número de interfaz de programación de aplicaciones (API) y *frameworks* que implementen por nosotros la estructura de la mayor parte de los tipos de redes y las funciones necesarias para trabajar con ellas. Uno de estos *frameworks* es el *Toolbox Neural Network* que *Matlab* posee, que ofrece una amplia variedad de arquitecturas y funciones de entrenamiento para modelar complejos sistemas no lineales de un modo sencillo utilizando RNA. Asimismo, soporta aprendizaje supervisado, no supervisado y reforzado utilizando arquitecturas como: *perceptrón*, *perceptrón* multicapa, *adaline*, base radial, mapas auto-organizados, redes competitivas, etc.

Matlab utiliza una estructura única que da acceso a todas las propiedades de la red neuronal, independientemente del tipo que esta sea, de manera que utilizando esta propiedad se puede modificar las entradas, capas, conexiones, pesos, etc. De esta manera una vez configurada la red neuronal según nuestras necesidades se puede invocar las funciones de manipulación de redes neuronales disponibles en *Matlab* (inicialización, entrenamiento, simulación, etc.), introduciendo en la línea de comandos códigos *Matlab* (Anexo 2).

Una vez creada la red neuronal, y luego trabajar con la misma red configurada, se puede utilizar las siguientes funciones principales, operando:

- Inicialización (*init*)
- Entrenamiento (*train*)
- Simulación (*sim*)

3.5. DESARROLLO DE MODELOS HIBRIDO WAVELET REDES NEURONALES

Wang y Ding (2003) indican que el modelo híbrido WRN hace uso de los méritos del análisis *wavelet* y RNA, por lo que tiene un excelente desempeño en la simulación y pronóstico. Asimismo, señalan que el modelo híbrido WRN es funcional y desempeña un papel importante en la mejora de la precisión y prolongar el horizonte de pronóstico o la hidrología, además, sugieren que es apropiada su aplicación a cualquier serie temporal de recursos hídricos.

En la presente sección se describe los procedimientos de construcción de la técnica combinada de análisis multiresolución *wavelet* y RNA en el pronóstico de series hidrológicas con el propósito de mejorar la precisión de los modelos de RNA solos. El enfoque híbrido, es decir, los modelos de la IA (tal como la RNA) combinada con el análisis multiresolución *wavelet* como una herramienta de preprocesamiento de series de tiempo, se aplicaron para el pronóstico de caudales diarios del río Amazonas a múltiples horizontes de pronóstico. Además, el desempeño de estos modelos híbridos WRN fueron evaluados mediante índices estadísticos, en la fase de entrenamiento y validación.

El desarrollo de este modelo híbrido se basó en varios estudios, como en “*River flow forecasting using an integrated approach of wavelet analysis and artificial neural networks*”, realizado por Badrzadeh y Sarukkalige (2012), que utilizaron un modelo híbrido para la predicción de caudales medias diarias de la estación Northam Weir, en el río Avon ubicado en Australia. En “*A combined generalized regression neural network wavelet model for monthly streamflow prediction*”, de Kisi (2011), que combinó dos métodos, transformada de *wavelet* discreta y red neuronal de regresión generalizada para la previsión de caudales mensuales de dos estaciones, Gerdelli en el río de Canakdere y Isakoy en el río de Goksudere, en Turquía.

También en “*A wavelet-neural network hybrid modelling approach for estimating and predicting river monthly flows*”, desarrollado por Wei *et al.* (2013), que utilizaron un

enfoque de modelamiento híbrido *wavelet* red neuronal para la estimación y predicción de caudales mensuales del río Weihe en China. En “*Daily crude oil price forecasting using hybridizing wavelet and artificial neural network model*”, de [Shabri y Samsudin \(2014\)](#), que proponen un nuevo método basado en la integración del modelo de transformada de *wavelet* discreta y RNA para el pronóstico diario de los precios del petróleo crudo.

Finalmente, en “*Predicting river daily flow using wavelet-artificial neural networks based on regression analyses in comparison with artificial neural networks and support vector machine models*”, de [Shafaei y Kisi \(2016\)](#), que investigaron la capacidad del modelo *wavelet* redes neuronales artificiales para la predicción de caudales diarios a corto plazo de la estación Vanyar, en el río Ajichai, en la región noroeste de Irán.

Por lo tanto, El modelo híbrido WRN es un modelo integrado de redes neuronales artificiales (RNA) con la entrada preprocesada por la transformada *wavelet* discreta (TWD). En otras palabras, este modelo híbrido incorpora dos submodelos principales de descomposición *wavelet* y RNA. La salida del submodelo *wavelet* se impone como entrada al submodelo de RNA. La salida del modelo híbrido no es descompuesto, para pronósticos N pasos adelante de la serie temporal de caudal. La Figura 32 ilustra la estructura del modelo híbrido *wavelet* red neuronal.

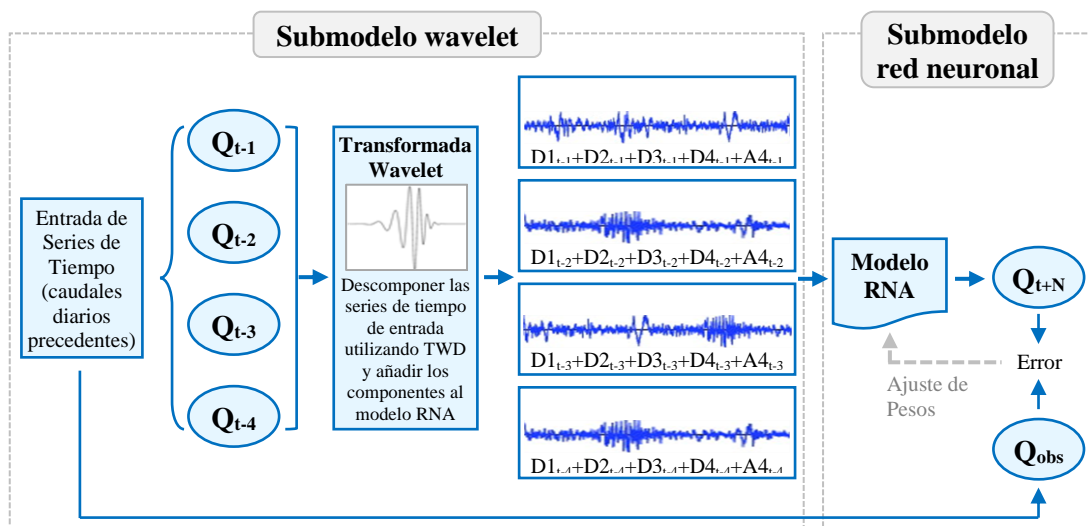


Figura 32: Estructura del modelo híbrido *wavelet* red neuronal (WRN) para pronósticos N pasos adelante

Para las entradas del modelo híbrido WRN, las series diarias de caudal actual y antecedentes se descomponen en un cierto número de componentes (subseries temporales: Aproximación- A_n y Detalles- D_n) utilizando el algoritmo de TWD ([Mallat 1989](#)). Cada

componente desempeña un papel diferente en la serie temporal original y el comportamiento de cada subserie temporal es distinta (Wang y Ding 2003). Es decir, el modelo híbrido WRN se construye, en la que las subseries A_n y D_n de la serie temporal original son entradas del RNA y se emiten la serie temporal original que son salidas del RNA, así estimar los valores futuros del caudal.

3.5.1. Submodelo *wavelet*

El análisis multiresolución *wavelet* se implementa como un submodelo (Figura 32). La serie temporal original se descompone en sus componentes de baja y alta frecuencia (Aproximación y Detalles). Este submodelo tiene la capacidad de aplicar diferentes *wavelets* madre y descomponer la señal en múltiples niveles. No hay una manera directa de determinar la *wavelet* madre o el nivel de descomposición óptima. Por lo tanto, antes de realizar la descomposición *wavelet*, se requiere seleccionar tres aspectos de parámetros del submodelo *wavelet*: la *wavelet* madre, su orden y el número de niveles de descomposición.

La característica más importante de cada *wavelet* madre es su forma. Así, la elección de la *wavelet* madre depende de los datos a analizar. En esta investigación, se seleccionó la función *wavelet Daubechies*, una de las familias de *wavelets* más ampliamente utilizados en la hidrología. En cambio, el nivel óptimo de descomposición se determina a través del proceso de prueba y error. Aunque, en este caso, el nivel de descomposición se estableció a través de la Ecuación (2.27), considerada como el nivel recomendable de descomposición. Los fundamentos de selección se encuentran en la sección de desarrollo de las directrices para los modelos híbridos WRN, ver la Figura 33.

Después de descomponer las señales de entrada a los coeficientes *wavelets* deseados, posteriormente, estas subseries se alimentan en el submodelo de RNA (*perceptrón* multicapa). Cada una de estas subseries tiene un papel diferente en la serie temporal y es importante mantenerlas todas como entradas al submodelo de RNA.

3.5.2. Submodelo redes neuronales

El submodelo redes neuronales de la Figura 32 desarrolladas en esta investigación son las redes neuronales de retropropagación (*backpropagation* en inglés) de alimentación hacia adelante (*feedforward* en inglés) de tres capas, que se utiliza con mayor frecuencia en ingeniería de recursos hídricos. Además, para mejorar el desempeño de modelamiento de la

red y reducir la probabilidad de que el proceso de entrenamiento se encuentre atrapado en un mínimo local, la normalización de los datos se aplicó en la entrada y salida de la RNA, en el rango de [-1, 1] debido a la función tangente hiperbólica seleccionada, utilizando la siguiente ecuación:

$$x_n = 2 * \frac{(x_i - x_{mín})}{(x_{máx} - x_{mín})} - 1 \quad (3.15)$$

donde x_n es el valor normalizado de x_i , x_{max} y x_{min} son medidas de valores máximos y mínimos de la serie temporal original.

Para evitar problemas de mínimos locales, los datos de entrada también se dividen en dos subseries de entrenamiento y validación. La proporción de la subserie de datos de entrenamiento y validación se puede definir como una entrada del modelo. Para verificar la precisión del modelo, la subserie de validación en este modelo es una serie de datos completamente independiente, que no tiene ningún papel en el entrenamiento de las redes, por lo que, al comprobar los criterios de desempeño para la subserie de datos de validación, se puede confirmar la confiabilidad del modelo.

Para lograr los entrenamientos óptimos de las RNA, en esta investigación, se utilizó el algoritmo de entrenamiento llamado *Levenberg-Marquardt*, seleccionado debido a que suele ser más rápido y más fiable que otras variantes de retropropagación, descritas en el Capítulo II de revisión literaria. Las funciones de transferencia para las capas oculta y de salida son la función tangente sigmoïdal hiperbólica y lineal, respectivamente. El número óptimo de neuronas ocultas, se determinó utilizando los mismos criterios utilizados para las RNA solos. Tener una mayor cantidad de neuronas ocultas, le da a la red la flexibilidad requerida para resolver problemas más complejos, mientras que tener demasiadas neuronas puede causar sobreajustes.

En esta investigación, inicialmente se definió un valor referencial promedio de números de neuronas ocultas, tomando en cuenta el volumen de datos de entrada y el número de neuronas de entrada. En la práctica el número de neuronas ocultas se determinó mediante el proceso de prueba y error. Iniciándose el ensayo con un valor mínimo cercano a la referencia, e incrementándose en cada ensayo, hasta alcanzar un máximo razonable. El criterio de detención de cada ensayo se establece como el número de épocas (*epochs* en inglés). Los principales pasos en el desarrollo de modelos de RNA se da en la Figura 30.

3.5.3. Directrices para los modelos híbridos WRN

En general, los factores a tomar en cuenta en la construcción y desempeño de los modelos híbridos WRN fueron esquematizados y sintetizados en la Figura 33.

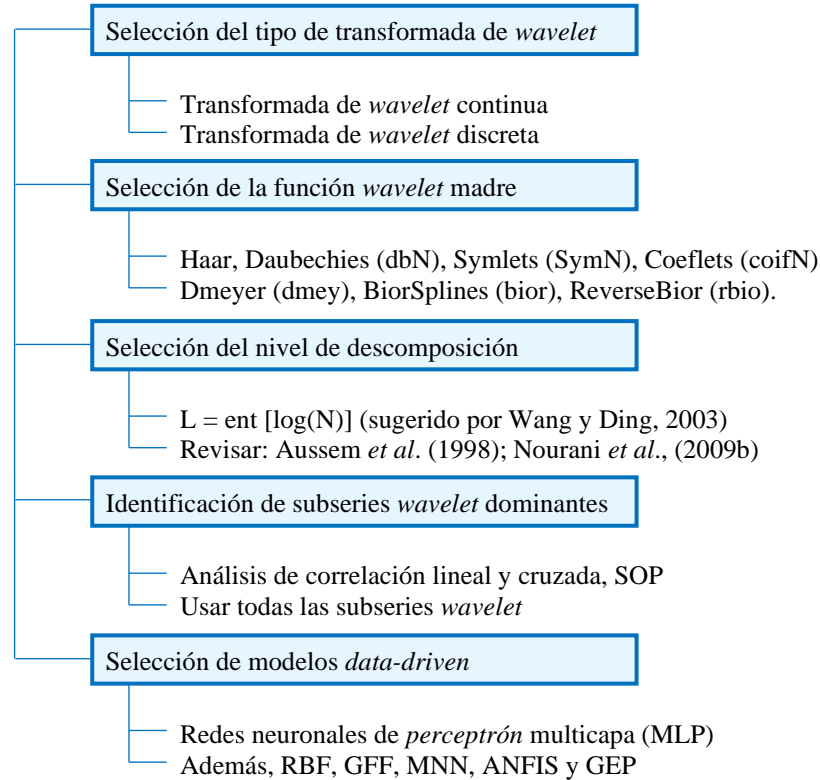


Figura 33: Principales pasos en el desarrollo de los modelos híbridos WRN

FUENTE: Esquematizado de [Shoaib et al. \(2016\)](#)

En resumen, las principales pautas generales y las opciones más utilizadas para la implementación exitosa de los modelos híbridos WRN, contiene la selección del tipo de transformada de *wavelet* (continua o discreta). Igualmente, el desempeño de los modelos híbridos WRN es también extremadamente sensible a la selección de la función *wavelet* apropiada de una serie de funciones *wavelet* disponibles. La eficiencia del performance de los modelos híbrido *wavelet* depende en gran medida de la selección del nivel de descomposición adecuado. Además, la elección del modelo “basados en datos” es otro factor vital ya que existen diferentes tipos de estos modelos disponibles, incluyendo la red neuronal de *perceptrones* multicapa. En consecuencia, a continuación, se detallan cada uno de estos factores seleccionados para la presente investigación.

a. Selección del tipo de transformada de *wavelet*

En hidrología, las series hidrológicas observadas se expresan a menudo como series discretas, por lo que la TWD se prefiere en la mayoría de los problemas de predicción de sistemas hidrológicos, debido a su simplicidad y capacidad para calcular con menos tiempo para descomponer una serie hidrológica en un conjunto de coeficientes y subseñales bajo diferentes escalas (Sang 2012).

En esta investigación, se utilizó la TWD para la descomposición de datos, debido a que la TWC calcula los coeficientes *wavelet* en todas las escalas posibles, lo que consume mucho tiempo y generan gran cantidad de información. Por el contrario, el TWD es más eficiente para el análisis porque limita el factor de escala y el factor de desplazamiento de la función *wavelet* básica a sólo valores discretos, mientras que se mantiene la precisión del análisis. La TWD se ha utilizado particularmente como una técnica computacional para extraer información sobre señales no estacionarias en los estudios en los que se basa el presente estudio, descritas en el Capítulo II de revisión literaria.

b. Selección de la función *wavelet*

Debido a la gran cantidad de familias de *wavelets* disponibles y cada uno contiene un número de funciones *wavelet* madre. En esta investigación, para descomponer la serie original se eligió la función *wavelet Daubechies* de orden 2 (db2), una de las familias de *wavelets* más ampliamente utilizados en la hidrología y orden 2 seleccionado por su gran desempeño en las pruebas realizadas previamente. Además, esta selección es reforzada con los estudios desarrollados por Nourani *et al.* (2011a), Singh (2011) y Maheswaran y Khosa (2012), que indican que la función *wavelet* db2 produce mejor performance que la función db1, db3, db4, *Sym2*, *Sym3*, *Sym4*, *Coif1* y *B-spline*.

c. Selección del nivel de descomposición

El número de neuronas de entrada para los modelos híbrido WRN, se estableció en función al mismo número de neuronas de entrada determinada para el modelo RNA que exhiben el valor más bajo de RMSE. Los cuales son incrementados de acuerdo al número de niveles de descomposición *wavelets* para ser empleados en los modelos WRN, mediante la Ecuación (2.27) planteada en los estudios de Aussem *et al.* (1998), Wang y Ding (2003) y Nourani *et al.* (2014a). De otro lado, el número máximo de niveles de descomposición *wavelets* depende

de la longitud de los datos disponibles. En suma, el número total de neuronas de entrada al modelo híbrido WRN están en función al número de entradas establecidas para las RNA y al número de niveles de descomposición óptima.

d. Selección del modelo “basados en datos”

Entre los modelos “basados en datos” (*data-driven* en inglés), la red neuronal de *perceptrón* multicapa (MLP) *feedforward* entre muchos paradigmas de RNA es de lejos, el más comúnmente utilizado en modelamiento hidrológico debido a su simplicidad, el cual fue reportado por [Maier y Dandy \(2000\)](#). Asimismo, se encuentra que el MLP es el tipo de red neural más ampliamente utilizado para el propósito y desarrollo de modelos de red neuronal acoplados a *wavelet* ([Shoaib et al. 2016](#)). Además, estos fueron desarrollados para el pronóstico de caudales a corto plazo, que por lo general utiliza la técnica de propagación de error hacia atrás para entrenar la configuración de la red ([Wu et al. 2009](#)). En las redes de alimentación hacia adelante (*feedforward* en inglés) las conexiones entre los nodos fluyen en una dirección: desde una capa de entrada, a través de una o más capas ocultas, a una capa de salida ([Dawson y Wilby 2001](#)).

e. Identificación de subseries *wavelet* dominantes

La identificación de subseries de datos *wavelets* que contienen información significativa sobre el sistema es también otro aspecto importante en la implementación exitosa de modelos hidrológicos acoplados a *wavelet*. Debido a que el uso de gran cantidad de entradas no sólo aumenta el tiempo de simulación, sino que también aumenta la complejidad computacional ([Shoaib et al. 2016](#)).

El enfoque de selección de subseries *wavelets* dominantes sobre la base de la correlación lineal es criticado por [Nourani et al. \(2011b, 2013\)](#), ya que puede existir una fuerte relación no lineal entre la entrada y salida objetivo (*target* en inglés) a pesar de la presencia de correlación lineal débil. Por lo tanto, es obvio, a partir de la literatura citada anteriormente, que, en esta investigación, se utilizaron todas las subseries *wavelet* como entrada al modelo híbrido WRN.

Por otro lado, los modelos híbridos WRN fueron calibrados con la misma subserie de entrenamiento empleados para los modelos tradicionales de RNA, a excepción de que las subseries de entrada al modelo híbrido se formularon a partir de las subseñales después de

aplicar el análisis de descomposición *wavelets*. El proceso de selección del número óptimo de neuronas en la capa oculta, fue similar al proceso de prueba y error empleado para los modelos de RNA solos.

3.5.4. Implementación del análisis *wavelet* en *Matlab*

Para muchas series temporales, el contenido de baja frecuencia es la parte más importante. Es lo que da a la serie su identidad. Por otro lado, el contenido de alta frecuencia imparte sabor o matiz. En el análisis multiresolución *wavelet*, a menudo hablamos de Aproximaciones y Detalles. Las Aproximaciones son los componentes de alta escala y baja frecuencia de la serie temporal, mientras, los Detalles son los componentes de baja escala y alta frecuencia.

La “*Wavelet Toolbox*” es una herramienta que proporciona funciones y aplicaciones para analizar y sintetizar señales e imágenes mediante *wavelets*. Se ejecuta dentro del entorno *Matlab* por lo que es necesario tener instalado previamente este programa, así como la *Toolbox* en el ordenador.

La implementación de la “*Wavelet Toolbox*” en *Matlab* para llevar a cabo el tratamiento de serie hidrológica por medio de la transformada de *wavelet*, incluye un gran número de *wavelets* que se puede utilizar tanto para análisis continuos como discretos. En específico para trabajar el análisis *wavelet* discreto unidimensional, utilizando la línea de comandos, se muestran de manera ampliada en el Anexo 3. El proceso de filtrado en esta investigación, en su nivel más básico (operaciones fundamentales), se utilizaron las siguientes funciones:

- Descomposición *wavelet* multinivel (*wavedec*)
- Reconstrucción de la Aproximación y los Detalles (*wrcoef*)

3.6. COMPARACIÓN DE LAS DOS TÉCNICAS DE LA IA

En esta investigación, las dos técnicas de modelamiento “basados en datos”, es decir, redes neuronales artificiales (RNA) e híbridos *wavelets* redes neuronales (WRN), se desarrollaron para el análisis comparativo en el pronóstico de caudales diarios de uno hasta treinta días de anticipación. Juzgar la efectividad de los modelos de pronóstico formulados es inevitable al seleccionar los mejores modelos. Es así que, al momento de seleccionar los mejores modelos

candidatos de cada técnica, se usaron medidas estadísticas de error (RMSE, MAE y MARE) y eficiencia (NSE), y análisis gráfico detallado de errores para evaluar su desempeño.

Una vez seleccionada la red con la estructura óptima en cada técnica y determinado el número máximo de vectores de entradas inmejorables al modelo RNA, y por ende al modelo híbrido WRN (o sea, el vector de entrada fue ampliado en función al nivel de descomposición *wavelet*). En seguida, el desempeño de los mejores modelos de RNA e híbridos WRN para pronósticos a un día de anticipación, se compararon utilizando dos índices de desempeño, es decir, RMSE y NSE, a través de su mejoramiento y/o empeoramiento, en por ciento, considerando el desempeño de la técnica de RNA como los modelos de referencia (*Benchmark* en inglés).

También, la comparación de las dos técnicas para los siguientes diez horizontes de pronóstico (sintetizado desde tres hasta 30 días de anticipación, es decir, Q_{t+3} , Q_{t+6} , Q_{t+9} , ..., hasta Q_{t+30}), se utilizaron los índices estadísticos RMSE, MAE, MARE y NSE, a través de su mejoramiento y/o empeoramiento, en por ciento, considerando como modelos de referencia (*Benchmark*) la técnica de RNA para múltiples horizontes de pronóstico.

3.6.1. Pronóstico a un día de anticipación

Para el análisis comparativo de las dos técnicas de pronóstico a un día de anticipación, se examinaron todos los modelos de RNA formulados, con estructuras de entrada de uno hasta la óptima. De igual modo, la misma cantidad de entradas originales fueron consideradas para los modelos híbridos WRN. Pero, estas entradas fueron preprocesadas mediante la descomposición *wavelet*, que ameritó evaluar el performance de estos modelos WRN en comparación a los modelos de RNA tradicionales. Por lo tanto, a través de la valoración de los índices estadísticos RMSE y NSE, se establece la mejora de la precisión de pronóstico, tanto en la fase de entrenamiento como en la validación.

En por ciento de aumento y/o disminución de los índices de desempeño del modelo híbrido WRN en comparación a los RNA referenciales, se califica como se describe en seguida, el signo negativo del RMSE en por ciento, indica una mejora del índice a través de su disminución, debido a que cuanto más cerca de 0.0, es el valor de RMSE, es mejor. El signo positivo del estadístico NSE en por ciento, muestra la mejora de este índice a través de su aumento, ya que cuanto más cerca de 1.0, es el valor de NSE, es mejor.

3.6.2. Pronóstico con horizontes más amplios

La finalidad de esta investigación fue conocer la capacidad predictiva de los modelos de la inteligencia artificial propuestos (RNA e híbrido WRN). Por lo tanto, en esta sección, se evaluaron los mejores modelos (redes con estructuras óptimas) de cada técnica de pronóstico. Los modelos seleccionados fueron utilizados para ampliar el horizonte de pronóstico, hasta un máximo de 30 días de anticipación (sintetizados de tres en tres), así, generándose diez modelos de pronóstico de cada técnica.

El propósito de la evaluación comparativa es identificar las brechas en por ciento de su desempeño o sus métricas entre ambas técnicas. De esta manera, establecer la mejora de la calidad de pronóstico de caudales diarios, durante las fases de entrenamiento y validación. Dicha evaluación consiste en el proceso de equiparar en por ciento el desempeño de los modelos híbridos WRN en referencia a los modelos de RNA para cada horizonte de pronóstico, a través de cuatro medidas estadísticas (RMSE, MAE, MARE y NSE). Además, es complementado con el análisis gráfico de la variación de los índices RMSE y NSE al aumentar el horizonte de pronóstico.

La diferencia en por ciento determinada para cada horizonte de pronóstico, los signos negativos del RMSE, MAE y MARE en por ciento, indican una mejora de los índices a través de su disminución, debido a que cuanto más cerca de 0.0, son los valores de RMSE, MAE y MARE, son mejores. El signo positivo del estadístico NSE en por ciento, muestra la mejora de este índice a través de su aumento, ya que cuanto más cerca de 1.0, es el valor de NSE, es mejor. De otro lado, según la evolución del performance de los modelos en términos de NSE son calificados de acuerdo a los criterios de [Dawson y Wilby \(2001\)](#).

IV. RESULTADOS Y DISCUSIÓN

Las RNA han venido siendo utilizadas en los últimos años para la resolución de problemas en el área de los recursos hídricos y recientemente en la modelación y pronóstico de caudales, como una alternativa a los modelos de series temporales lineales (Zhang *et al.* 1998; Maier y Dandy 2000; Mishra y Desai 2006; Adamowski 2008; Wang *et al.* 2009). En esta investigación, se desarrollaron una serie de modelos “basados en datos” de pronóstico de caudales diarios con dos técnicas de la inteligencia artificial para múltiples horizontes de pronóstico (de uno hasta 30 días de anticipación), dando lugar a modelos de RNA y un enfoque híbrido *wavelet* red neuronal (WRN), cuyas metodologías fueron descritas en el Capítulo anterior.

A lo largo de esta sección, se utilizó una misma serie de datos previamente analizados para identificar los dos grupos de modelos de RNA e híbridos WRN, que logren captar el proceso que sigue los caudales diarios del río Amazonas (estación hidrométrica de Tamshiyacu, en Perú), en el periodo de 1985 al 2012. La finalidad del desarrollo de esta sección fue el de realizar una comparación efectiva e imparcial entre los métodos de RNA y los modelos híbridos WRN, considerando los mismos patrones de entrada y salida para ambos, y no el de innovar estas técnicas. El trabajo numérico de los modelos de RNA e híbridos WRN se desarrollaron en el entorno del *software Matlab R2017a* y sus herramientas *Toolboxes Wavelet™* y *Neural Network™*.

Por otro lado, los resultados de la evaluación de los modelos de RNA a través de los criterios de eficiencia, fueron considerados como un método de referencia (*benchmark* en inglés) para comparar el desempeño de los modelos híbridos WRN, por ende, permitió conocer la mejora o empeoramiento de sus índices estadísticos. Además, estos modelos fueron calificados en base a la magnitud de sus errores encontrados en ambas fases. Obviamente, se calcularon para una serie de datos de validación totalmente diferentes de aquellos utilizados durante la fase de entrenamiento. Cuyo propósito fue de identificar el mejor ajuste y la mejor técnica para el problema de pronóstico.

4.1. ANÁLISIS PRELIMINAR DE DATOS

En esta sección, se presenta los resultados del análisis cualitativo y cuantitativo de la serie de caudales diarios, utilizando herramientas de la estadística clásica y métodos gráficos (Análisis exploratorio de datos, *EDA* por sus siglas en inglés), que ayudaron a identificar las características relevantes que explican rasgos del comportamiento de la serie. Los datos analizados consisten de una serie de 27 años de registro que fueron proporcionados por el Servicio Nacional de Meteorología e Hidrología (SENAMHI) del Perú, como parte del observatorio SO-HYBAM (anteriormente ORE-HYBAM, www.ore-hybam.org). Los datos corresponden a caudales medios diarios registrados en la estación hidrométrica de Tamshiyacu (río Amazonas) en el periodo comprendido entre el 01 de setiembre del 1985 al 31 de agosto del 2012, siendo un total de 9 862 muestras.

La identificación de la arquitectura óptima de una RNA requiere del análisis exploratorio previo del conjunto de datos a utilizar. Faraway y Chatfield (1998) recomiendan que un buen modelo de RNA para series de datos temporales debe ser seleccionado combinando habilidades tradicionales de modelación con conocimientos de análisis de series temporales y de los problemas particulares. En esta sección, se presenta los resultados del análisis preliminar de datos (tendencia, estacionariedad, estacionalidad, máximos y mínimos, etc.), para el conocimiento de la serie y argumentos desde los cuales partir para la identificación de una o varias redes neuronales.

4.1.1. Estadísticos básicos y patrones predominantes

a. Tendencia

Con el fin de observar los patrones predominantes en la serie de caudales del río Amazonas en la estación hidrométrica de Tamshiyacu, se ha graficado la serie completa en la Figura 34. Al agregar una línea de tendencia se observa que esta refleja una pendiente de $0.21804 \text{ m}^3\text{s}^{-1}/\text{día}$ y una magnitud de la tendencia decreciente a largo plazo. Además, en el mismo gráfico se observa que existe un patrón horizontal (estacional), es decir, los valores oscilan a través del tiempo alrededor de la línea de tendencia, aunque cada uno de estos máximos y mínimos con distinta magnitud.

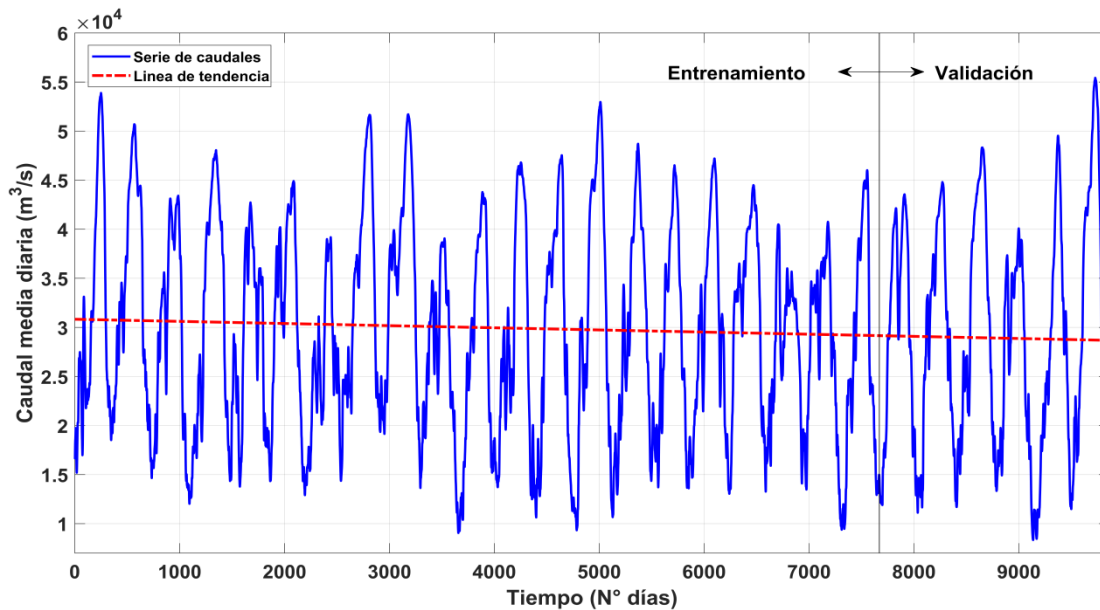


Figura 34: Serie temporal de caudales diarios del río Amazonas (Tamshiyacu) de 01 de setiembre de 1985 a 31 de agosto de 2012. Línea de tendencia en rojo

En la Tabla 1 se resume los estadísticos básicos de la serie de caudales del río Amazonas, serie completa (1985-2012) y también de cada una de las subseries de datos segmentados (entrenamiento y validación, Figura 34).

Tabla 1: Resumen de estadísticos básicos de la serie de caudales diarios total, subseries de entrenamiento y validación del río Amazonas

Estadístico	Total 1985-2012	Entrenamiento 1985-2006	Validación 2006-2012
Número de datos	9 862	7 670	2 192
Valor medio (m ³ /s)	29 750	29 910	29 180
Mediana (m ³ /s)	29 340	29 560	28 840
Desv. Est. (m ³ /s)	10 870.90	10 635.53	11 641.98
Varianza (m ³ /s) ²	0.3654	0.3556	0.3990
Mínimo (m ³ /s)	8 327	9 046	8 327
Máximo (m ³ /s)	55 420	53 880	55 420
Interango (m ³ /s)	17 952.5	17 340	20 210
Asimetría	0.1037	0.0999	0.1377
Curtosis	-1.0020	-1.0011	-1.0430

b. Estacionariedad

No obstante, un concepto intuitivo de una serie estacionaria se cumpliría si no existiese en la serie un cambio sistemático en la media (tendencia) y si no existiese una variación

sistemática de la varianza. En otras palabras, las propiedades de una sección de datos son muy similares a cualquier otra sección de la serie.

Para saber si una serie temporal es o no estacionaria, existe una serie de pruebas como la representación gráfica y la función de autocorrelación (FAC), que son las más usadas y que fueron de utilidad para la presente investigación. Asimismo, las series no estacionarias pueden ser detectadas calculando el coeficiente de la función de autocorrelación definido por [Box et al. \(1976\)](#).

$$r_k = \frac{(\sum(x_t \cdot x_{(t+k)})) - \left(\frac{1}{n-k} \sum x_t \cdot \sum x_{(t+k)}\right)}{\left[\sum x_t^2 - \frac{1}{n-k} (\sum x_t)^2\right]^{1/2} \left[\sum x_{(t+k)}^2 - \frac{1}{n-k} (\sum x_{(t+k)})^2\right]^{1/2}} \quad (4.1)$$

donde x es la variable, t es el tiempo y las sumatorias se realizan desde $t = 1$ a $t = n - k$. La ecuación (4.1) determina el grado de correlación entre observaciones que están separados k unidades de tiempo.

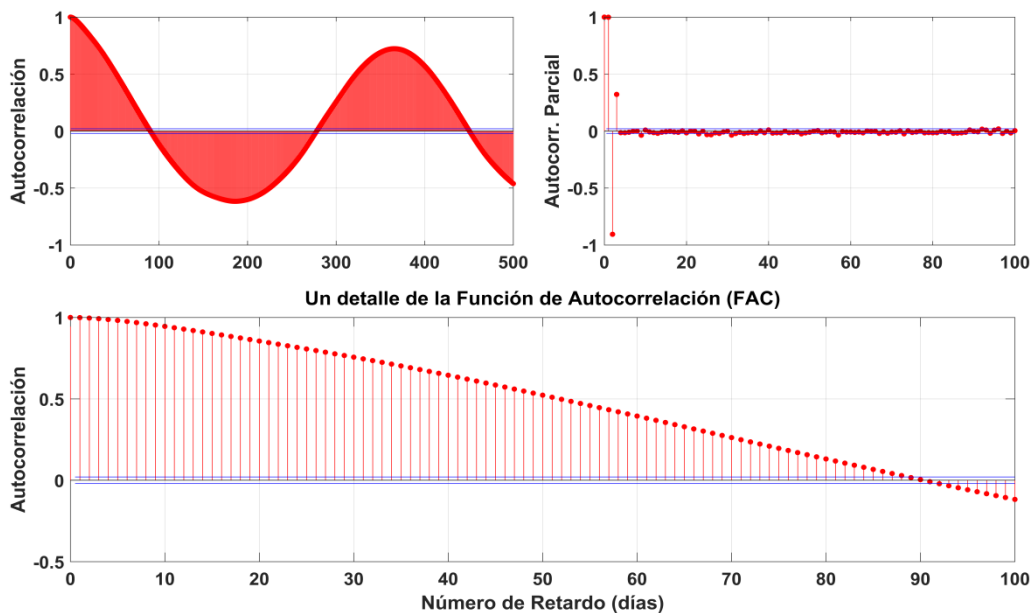


Figura 35: Gráficos de la FAC y FACP de caudales de 1985 al 2012 (superior), detalle de la FAC para los primeros 100 retardos (inferior). Las líneas azules competen al umbral de significancia estadística al nivel del 95 por ciento

La Figura 35 representa el correlograma o (FAC) y FAC parcial (FACP) de la serie de caudales del río Amazonas (estación hidrométrica Tamshiyacu) para los primeros 500 retardos y 100 retardos, respectivamente. Se puede observar que existe una fuerte dependencia con los valores anteriores o lo que es lo mismo, los coeficientes de r_k no se reducen rápidamente hacia cero, que es el comportamiento del correlograma de una serie no

estacionaria. Por lo tanto, se considera a la serie temporal como no estacionaria debido a efectos estacionales. La interpretación del correlograma tiene más usos que el que se le ha dado en esta sección; así como el que fue de utilidad a la hora de identificar un modelo adecuado para la serie de caudales posteriormente.

c. Variabilidad del régimen de caudales

La zona oeste de la cuenca Amazónica en Perú, experimenta una alta variabilidad espacial de la precipitación, la parte norte de la cuenca muestra mayor precipitación y de forma constante, generando una débil estacionalidad, mientras hacia la parte sur, se tiene un claro régimen tropical con una fuerte estacionalidad (periodo lluvioso en el verano austral entre los meses de diciembre a marzo y periodo más seco en el invierno austral), asociada con la migración estacional de la zona de convergencia intertropical (ZCIT) y el monzón sudamericano, así lo manifiesta [Espinoza et al. \(2009b\)](#).

En la planicie de la amazonía peruana (estación Tamshiyacu), el caudal medio mensual revela una fluctuación anual unimodal regular, comenzando el ciclo hidrológico anual en agosto para las lluvias y en octubre para los caudales (el desfase de dos meses entre las precipitaciones y caudales corresponde al tiempo de concentración de la cuenca). Así se observan descargas máximas ($\sim 44\,500\text{ m}^3/\text{s}$) durante abril y mayo, y las mínimas ($\sim 15\,000\text{ m}^3/\text{s}$) en los meses de agosto a setiembre, generándose caudales máximos que son tres veces mayores que los mínimos y una fluctuación de nivel de agua de siete metros ([Espinoza et al. 2006, 2011a, 2013; Ronchail et al. 2018](#)). Por ende, es obvio que los caudales máximos son más hostiles debido a que generan riesgos de inundación. Aunque, ambos eventos extremos igualmente implican decisiones de gestión y operación del sistema con el fin de mitigarlos.

La Figura 36 muestra la variación del régimen de caudales mensuales en la estación Tamshiyacu, estos componentes periódicos que se repiten varias veces en el año fue elaborado a través de dos técnicas señalado por [Chatfield \(2001\)](#), es decir, inicialmente identificado con el autocorelograma (dominio del tiempo, Figura 35) y en sucesivo elaborado mediante el uso del dominio de las frecuencias mensuales en el ciclo anual. Adicionalmente, de la aplicación del *boxplot* en cada mes del año (de enero a diciembre), integrado con la media mensual (cuadros negros), se observa un patrón estacional similar en todos los años con valores mínimos en los dos tercios del año (agosto-setiembre) y

valores máximos en un tercio del año (abril-mayo), aunque cada uno de estos máximos y mínimos con distintas magnitudes.

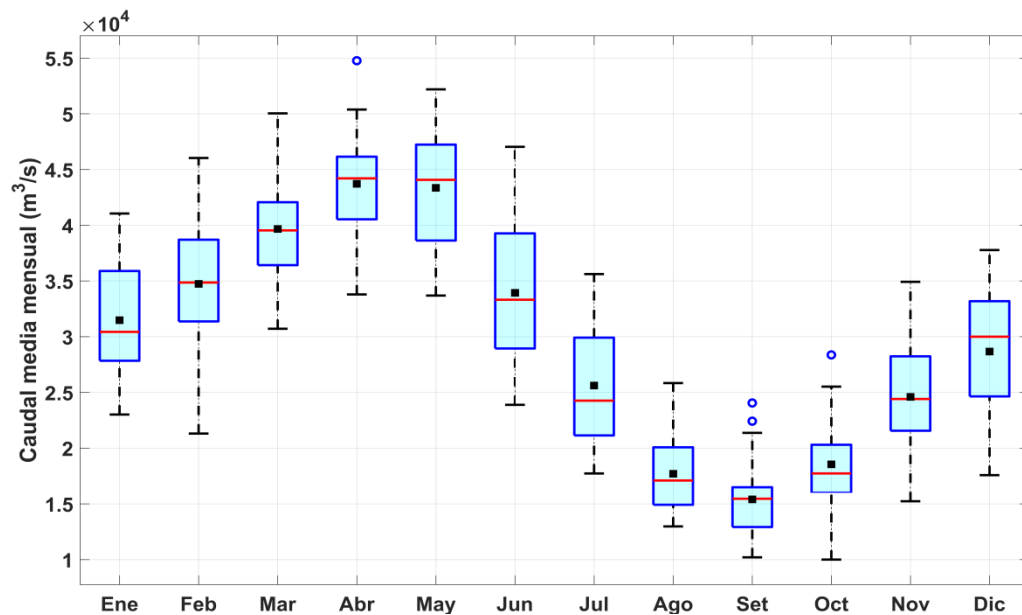


Figura 36: Variación del régimen de caudales mensuales en la estación Tamshiyacu. Cuadros negros indican la media mensual para cada mes del año

4.2. MODELOS BASADOS EN REDES NEURONALES ARTIFICIALES

Un modelo de pronóstico de caudales diarios basado en redes neuronales considera a la serie de caudales como una serie temporal $Q_t (t = 1, 2, \dots, n)$, y se intenta estimar los valores futuros, Q_{n+1} , a partir de la información que conllevan los valores del pasado. Se obtiene la función $f(Q \cdot w_j - b_j)$ que relaciona el caudal en el instante $n + 1$ con el caudal de los instantes $1, 2, \dots, n$.

Ya se ha mencionado en otras secciones que el buen desempeño de las RNA radica en su capacidad de entrenar las relaciones entre las entradas (datos históricos de los caudales) y sus salidas (caudal observada), para después generalizar y predecir el caudal dentro de la serie de datos observados. Es sumamente importante la correcta elección de vectores de entrada a la red, ya que es conocido que impacta directamente en la eficiencia de los pronósticos de los modelos de redes neuronales. No menos importante resultan ser la topología de la red, el algoritmo de entrenamiento y en una menor medida lo es también el número de iteraciones.

Por lo tanto, es necesario determinar los inputs o entradas, su preprocesamiento o transformación antes de su presentación a la red y su postprocesamiento o transformación

de los datos de salida de la red, así como la arquitectura de la red (número de capas ocultas, número de neuronas en cada capa oculta, conexiones entre capas, función de activación, etc.) y el algoritmo de aprendizaje o entrenamiento.

4.2.1. Segmentación y preprocesamiento de datos

La serie de caudales diarios con la que se contó corresponde a un total de 27 años (9 862 muestras) de registros históricos, desde 01 setiembre de 1985 a 31 agosto de 2012. La serie de datos se ha dividido en dos subseries de entrenamiento y validación (Figura 34 y Tabla 1), y se han utilizado 21 años de 01 setiembre de 1985 a 31 agosto de 2006 (7 670 muestras, 78.8 por ciento) para el entrenamiento de la red y seis años de 01 setiembre de 2006 a 31 agosto de 2012 (2 192 muestras, 22.2 por ciento) se ha reservado para validar la red entrenada y analizar sus desempeños. Los registros de caudal diario han sido preprocesados y escalados para que los valores caigan en el intervalo de $[-1,1]$ y estén acorde con la función tangente sigmoïdal hiperbólica que se ha utilizado en la capa oculta de los modelos de redes neuronales formuladas (Figura 37).

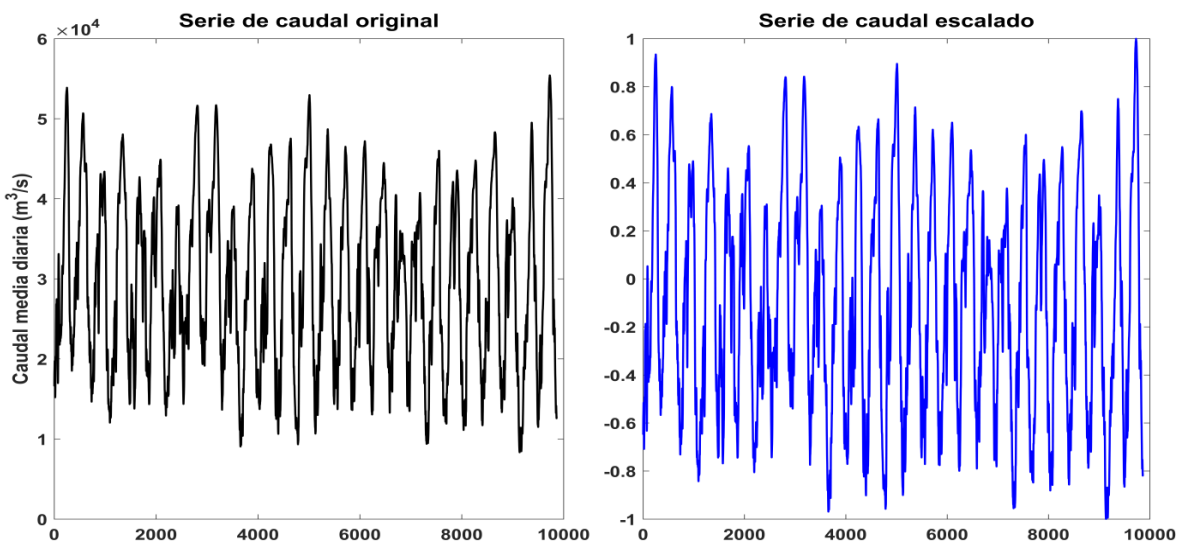


Figura 37: Serie de caudal diario de registros históricos original (izquierda) y escalada en el rango $[-1,1]$ (derecha)

4.2.2. Patrones de entrada a las RNA

El poder de un modelo de RNA se deriva de su capacidad de mapear patrones de entrada a patrones de salida siempre que existan las relaciones entre ellos. Es obvio que las relaciones entre las entradas y la salida son muy importantes para que un modelo funcione correctamente. En esta investigación, por tratarse de modelos de RNA univariados, la única

variable de entrada a la red considerada para entrenar fueron los rezagos de la misma serie temporal, además, considerando una sola salida objetivo sugerido en hidrología.

No hay formas matemáticamente rigurosas de seleccionar los patrones de entrada apropiadas. En esta investigación, el conocimiento del proceso hidrológico y el análisis de autocorrelación (Figura 35) se emplearon para seleccionar las entradas a la red que podría representar mejor el caudal actual. Sin embargo, de la FAC se observan aproximadamente 87 rezagos significativos; lo que se considera una cantidad muy elevada de entradas para el modelo, ya que se necesitarían 87 valores pasados del caudal para poder predecir un valor (el valor siguiente). Por esta razón, fue preciso llevar a cabo ensayos de prueba y error para determinar un menor y más prudente número de entradas.

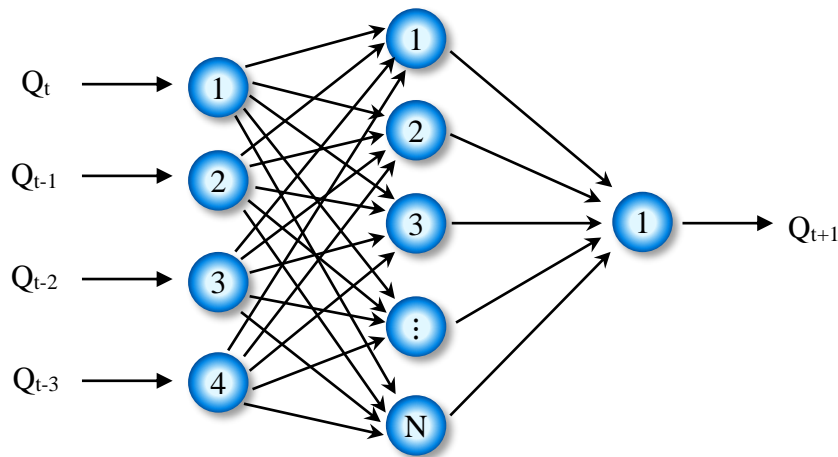


Figura 38: Modelo RNA4 (4-N-1) con cuatro entradas, N nodos ocultos y una salida

De otro lado, se tuvo claro que los valores de caudal en el instante Q_{t+1} están fuertemente relacionados con el valor de caudal en el instante Q_t , Q_{t-1} , Q_{t-2} , así como al instante Q_{t-n} (por ejemplo, Q_{t+1} está relacionado con Q_t , Q_{t-1} , Q_{t-2} y Q_{t-3} (Figura 38 y la Tabla 2). Con estos datos se tuvo un vector de entrada de longitud 1, 2, 3 hasta n respectivamente, formado por el valor de caudal del día actual y días antecedentes. El valor de n fue determinado mediante ensayos de prueba y error.

El proceso de estructuración, se inició con modelos RNA de estructuras muy simples, por ejemplo, considerando únicamente el caudal del día actual para predecir el día posterior, ver modelo RNA1 de la Tabla 2. Posteriormente, se integraron a este vector, información de más días antecedentes para elaborar modelos de RNA con estructuras más complejas, hasta un vector de siete entradas a la red (siete días de la semana). La Tabla 2 resume la variación de los patrones de entrada y salida que se utilizaron en esta investigación. La

diferencia principal entre ellos radica en la dimensión del vector de entrada a la red, a través del cual incorpora valores de caudales de días antecedentes secuencialmente, pero conservando permanente el vector de salida.

Tabla 2: Patrones de entrada y salida de los modelos RNA para pronósticos de caudales a un día de anticipación

Modelo	Entradas	Días adelante	Nº Entradas	Salidas
RNA1	Q_t	1 día	1	Q_{t+1}
RNA2	Q_t y Q_{t-1}	1 día	2	Q_{t+1}
RNA3	Q_t , Q_{t-1} y Q_{t-2}	1 día	3	Q_{t+1}
RNA4	Q_t , Q_{t-1} , Q_{t-2} y Q_{t-3}	1 día	4	Q_{t+1}
RNA5	Q_t , Q_{t-1} , Q_{t-2} , Q_{t-3} y Q_{t-4}	1 día	5	Q_{t+1}
RNA6	Q_t , Q_{t-1} , Q_{t-2} , Q_{t-3} , Q_{t-4} y Q_{t-5}	1 día	6	Q_{t+1}
RNA7	Q_t , Q_{t-1} , Q_{t-2} , Q_{t-3} , Q_{t-4} , Q_{t-5} y Q_{t-6}	1 día	7	Q_{t+1}

4.2.3. Modelos de RNA para pronóstico a un día de anticipación

El entrenamiento de modelos de RNA es un proceso de ajuste de los pesos de la red. El proceso de entrenamiento trata de encontrar un conjunto de parámetros (pesos) que hace que los caudales simulados coincidan con los caudales observados lo más cerca posible. En una aplicación de RNA, el entrenamiento también es un proceso de determinación de la arquitectura de la RNA en términos de número de nodos ocultos. El algoritmo de entrenamiento utilizado en todos los casos fue el de *Levenberg-Marquardt*, por ser el método más rápido para entrenar redes neuronales *feedforward* de retropropagación de errores con una capa oculta de tamaño moderado, aunque requiere una mayor cantidad de memoria que otros algoritmos.

En la etapa de entrenamiento como la primera fase del proceso de pronóstico, el objetivo fundamental fue seleccionar el número de neuronas en la capa oculta (es decir, obtener el valor de N de la Figura 38) que presente un mejor desempeño del modelo. Tradicionalmente, el número de nodos ocultos para los modelos de RNA, se seleccionaba a través del método de prueba y error. Sin embargo, un estudio realizado por [Belayneh y Adamowski \(2012\)](#) indican que el mejor desempeño de una red neural se produce cuando el número de nodos ocultos es igual a $\log(m)$, donde m es el número de muestras de entrenamiento. Asimismo, indican que el número óptimo de neuronas ocultas es $2p+1$,

donde p es el número de nodos de entrada. En esta investigación, estas dos técnicas fueron de ayuda al momento de selección el número de nodos ocultos.

Además, en esta investigación, el entrenamiento de la RNA también se usó para encontrar la mejor combinación de patrones de entrada por medio del análisis de sensibilidad. El análisis de sensibilidad se centró en elegir el número de vectores de entradas con los caudales de días antecedentes. Para una combinación fija de vectores de entrada, se entrenaron las RNA con diferentes nodos ocultos y se calculó el desempeño de la RNA usando la serie de datos de entrenamiento y validación en términos de RMSE, MAE, MARE y NSE.

Como un primer intento, se entrenó el modelo RNA1 ($1-N-1$), o lo que es lo mismo, un vector de una entrada, N neuronas en la capa oculta y una neurona en la capa de salida que es un vector que contiene los datos objetivo correspondiente para pronóstico a un día de anticipación. Luego, a esta red se le evaluó su desempeño en la fase de entrenamiento y validación. La topología óptima conseguida fue de $N=1$ (es decir, 1-1-1), los estadísticos de desempeño en ambas fases, se presentan en la primera fila de la Tabla 3.

Una vez comprobada los resultados de la red anterior, se entrenó el modelo RNA2, con un vector de 2 entradas, N neuronas en la capa oculta y una neurona en la capa de salida. Como se observa en la segunda fila de la Tabla 3, la red optimizada (2-4-1) con 4 nodos ocultos tiene mejor desempeño cuando el número de vector de entrada aumenta a dos, destacándose una mejora de los estadísticos en ambas fases del proceso (entrenamiento y validación), de hecho, esperable.

Seguidamente, mientras se fue aumentando la cantidad del vector de entrada el desempeño de los modelos propuestos fue mejorando obviamente, tanto en la fase de calibración como en la validación. Sin embargo, los modelos RNA5, RNA6 y RNA7 no superaron el desempeño adquirido por el modelo RNA4, de hecho, no esperable. Por lo tanto, en aras de la simplicidad y una mejor capacidad de generalización, se cree que el modelo RNA4 (4-11-1) funciona mejor con cuatro vectores de entrada, 11 neuronas ocultas y una neurona como salida (Tabla 3).

El resumen de las estadísticas de desempeño (RMSE, MAE, MARE y NSE) para el periodo de entrenamiento y validación, conseguidos con los diferentes modelos de RNA optimizadas para pronósticos a un día de anticipación, se presenta en la Tabla 3.

Tabla 3: Resumen de desempeño de los modelos RNA para pronósticos a un día de anticipación, en la fase de entrenamiento y validación

Modelo	Estruc. RNA	Entrenamiento				Validación			
		RMSE	MAE	MARE	NSE	RMSE	MAE	MARE	NSE
RNA1	1-1-1	452.4876	338.5649	1.3572	0.99819	433.2804	330.6609	1.4038	0.99861
RNA2	2-4-1	190.9740	133.2795	0.5603	0.99968	179.3247	127.2072	0.5701	0.99976
RNA3	3-6-1	177.2795	122.1988	0.5077	0.99972	158.3611	113.2732	0.4998	0.99981
RNA4	4-11-1	176.3848	121.8851	0.5062	0.99972	157.6686	113.0903	0.4969	0.99982
RNA5	5-12-1	179.5280	123.4467	0.5132	0.99971	159.2558	114.5368	0.5038	0.99981
RNA6	6-7-1	180.5543	123.7843	0.5144	0.99971	158.2834	113.3793	0.4969	0.99981
RNA7	7-10-1	177.6517	122.8303	0.5112	0.99972	157.6872	113.3129	0.5007	0.99982

Nota: RMSE y MAE en unidades (m³/s). MARE en por ciento (%)

Para un mejor análisis de los resultados, se debe conocer que los primeros tres estadísticos (RMSE, MAE, MARE) miden la magnitud de los errores, por lo que interesa que sus valores sean mínimos. El último estadístico NSE determina la magnitud relativa de la varianza residual en comparación con la varianza de datos medidos, entonces un buen modelo será el que entregue valores cercanos a 1. Hasta este punto se ha utilizado estadísticos estándares, es decir, estadísticos más comunes que indican la bondad de ajuste para evaluar un mejor o peor desempeño de los modelos. Sin embargo, tomar una decisión basándose solamente en estos estadísticos podría llevar a cometer un error a la hora de seleccionar el mejor modelo.

Por lo comentado en el punto anterior, a la hora de seleccionar el mejor modelo surgió la necesidad de utilizar otros criterios alternativos. En esta investigación, para realizar un análisis más profundo de los errores, se utilizó los gráficos de autocorelograma (FAC), histograma y periodograma acumulada. La FAC aporta valores de las autocorrelaciones de los residuos. La información del histograma proporciona la frecuencia de los valores representados y su distribución. Además, el periodograma acumulativo brinda una evidencia gráfica de la normalidad de los residuos.

Se comienza señalando que, si bien la incertidumbre del modelo es realmente la desviación de un valor calculado del valor verdadero, el análisis que se sigue más bien se enfoca en la desviación del valor calculado de la observación. El error del modelo, ε_t , se define mediante la ecuación 4.2:

$$\varepsilon_t = Q_{obs,t} - Q_{sim,t}, \quad t = 1, 2, \dots, T \quad (4.2)$$

donde Q_{obs} es el caudal observado; Q_{sim} es el caudal calculado por el modelo; t es el índice de tiempo (días); y T es la longitud del tiempo de registro. Cuanto menor es el error, más cerca está el pronóstico del valor real.

En la Figura 39 parte superior, se muestra la serie de errores de pronóstico del modelo RNA4 para la fase de calibración y validación, cuyos valores se mantienen inferiores a 1 000 m³/s salvo en momentos puntuales. Además, del análisis más detallado de los errores de la fase de validación, se encontró que el aspecto de la FAC de errores (Figuras 39 inferior izquierda) muestra que los errores tienen la apariencia de un ruido blanco y no conservan algún rastro de autocorrelación de retardos periódicos, solamente se aprecia una ligera autocorrelación en los retardos de 1, 4, 7, 28 y 50 días. También, el histograma (Figuras 39 inferior centro) muestra que la mayoría de los errores se reúnen cerca a cero y se dispersan desde casi -500 m³/s hasta 500 m³/s, ajustándose a una distribución normal.

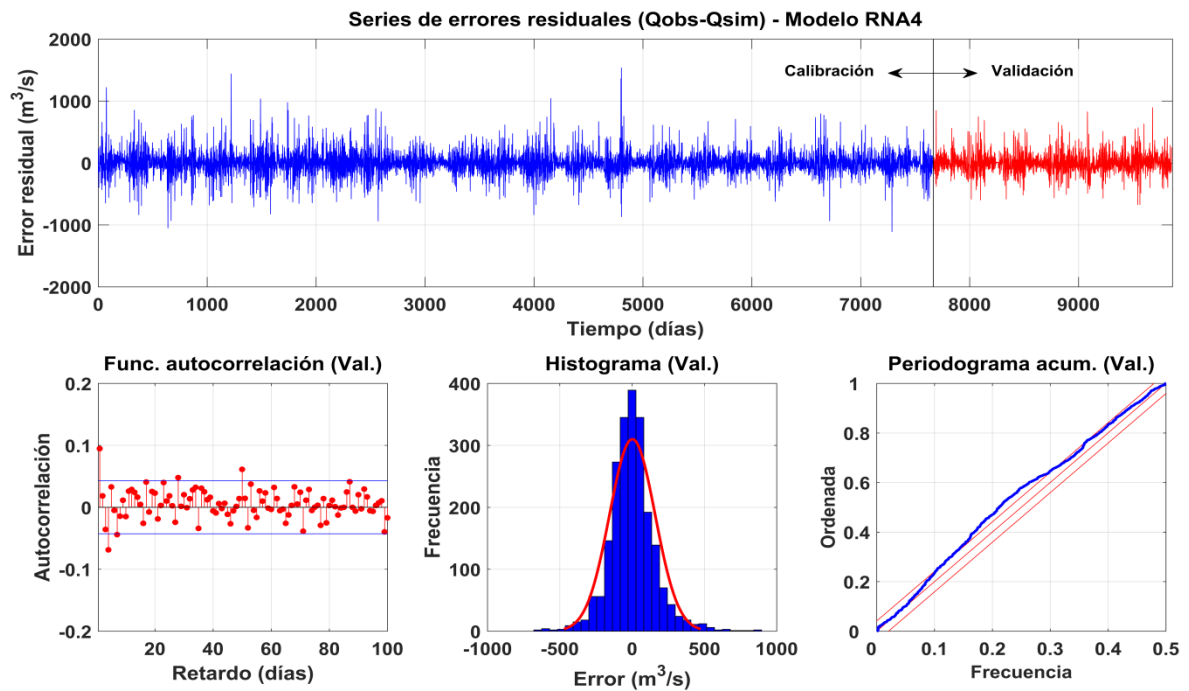


Figura 39: Serie de errores residuales del modelo RNA4 en la fase de calibración y validación (superior). Función de autocorrelación, histograma y periodograma acumulada en la fase de validación (inferior)

Según los estadísticos, los errores obtenidos por el modelo RNA4 son aceptables en la fase de entrenamiento e incluso mejoran (se reducen los errores) en la fase de validación. Sin embargo, el gráfico de la FAC indica que al menos 5 de los 100 coeficientes de correlación estimados son ligeramente significativos para el intervalo de confianza del 95 por ciento, por lo que los residuos no son completamente un ruido blanco como se esperaría si el modelo

RNA4 captara completamente la estructura de la serie. Esta aseveración se ratifica si se observa el periodograma acumulado (Figura 39 inferior derecha), que indica que las frecuencias acumuladas de los residuos superan los límites críticos del 95 por ciento que contendrían a las frecuencias si los residuos fueran un ruido blanco.

a. Comparación del desempeño de las RNA

A lo largo de esta sección, se han evaluado siete modelos de RNA distintos. Se diferencian entre sí tanto por la dimensión del vector de entradas, como por la cantidad de neuronas óptimas incluidas en la capa oculta. Se utilizaron vectores de entrada de dimensión 1, 2, 3, 4, 5, 6 y 7, con valores de caudal actual y antecedentes. Los resultados de desempeño de todas las RNA se resumen en la Tabla 3.

Se inició con la red más sencilla, es decir, el modelo RNA1 en el cual se utilizó solamente el valor del caudal del día actual para predecir el caudal del día siguiente. Luego del proceso de evaluación, se obtuvo un desempeño de $RMSE = 452.4876 \text{ m}^3/\text{s}$ y $433.2804 \text{ m}^3/\text{s}$; $MAE = 338.5649 \text{ m}^3/\text{s}$ y $330.6609 \text{ m}^3/\text{s}$, $MARE = 1.3572$ por ciento y 1.4038 por ciento; y $NSE = 0.99819$ y 0.99861 , durante la fase de entrenamiento y la validación, respectivamente.

La siguiente red evaluada fue la RNA2 que incorpora en su vector de entradas valores de caudal de los días actual y pasado. Con este nuevo modelo se obtuvo un desempeño de $RMSE = 179.3247 \text{ m}^3/\text{s}$, $MAE = 127.2072 \text{ m}^3/\text{s}$, $MARE = 0.5701$ por ciento y un $NSE = 0.99976$ en la fase de validación, mejorando notablemente a la red RNA1, no sólo en términos de eficiencia de *Nash-Sutcliffe* sino también en lo que respecta a los estadísticos $RMSE$, MAE y $MARE$. Asimismo, los modelos RNA3 y RNA4 mejoraron claramente a las redes anteriores en términos de los cuatro estadísticos utilizados, precisando que el modelo RNA4 se comportó mejor que la red RNA3.

Posteriormente, se entrenaron redes en las cuales se proporcionó más neuronas en la capa de entrada (RNA5, RNA6 y RNA7), es decir, valores del cuarto, quinto y sexto día antecedente, respectivamente; sin observar mejoras con respecto al modelo RNA4 (es decir, $RMSE = 176.3848 \text{ m}^3/\text{s}$ y $157.6686 \text{ m}^3/\text{s}$; $MAE = 121.8851 \text{ m}^3/\text{s}$ y $113.0903 \text{ m}^3/\text{s}$; $MARE = 0.5062$ por ciento y 0.4969 por ciento; y $NSE = 0.99972$ y 0.99982 , en la fase de entrenamiento y validación, respectivamente) que justificasen su inclusión, más bien al contrario, los errores se volvieron más altos.

Se puede concluir que, ninguno de los modelos de RNA desarrollados en esta investigación logró mejorar el desempeño que obtuvo el modelo RNA4 (4-11-1), en términos de los cuatro estadísticos (RMSE, MAE, MARE y NSE), ver en negrita en la Tabla 3. O sea que, la inclusión de los valores de caudal del día actual y los tres días antecedentes, aportaron información relevante a la RNA, resultando en mejoras evidentes en los pronósticos obtenidos, minimizando los errores residuales. Además, se corroboró con el análisis gráfico de errores; FAC de residuos (correlaciones ligeramente significativas en cinco retardos) y periodograma acumulativo de errores (mejor ajuste de las frecuencias acumuladas de los residuos dentro de los límites críticos del 95 por ciento). Sin embargo, no ocurre lo mismo con la inclusión de los caudales del cuarto, quinto y sexto día antecedente, resultando en un mayor esfuerzo computacional por lo que no se justifica su inclusión.

b. Modelo seleccionado de RNA

En las secciones anteriores se desarrolló modelos de RNA con distintas estructuras y se comparó sus desempeños en base a cuatro criterios estadísticos (RMSE, MAE, MARE y NSE). Además, se ha dicho que el modelo RNA4 ha obtenido los mejores valores de desempeño, es decir, un modelo con calificativo muy satisfactorio ($NSE > 0.90$; Dawson y Wilby 2001), el cual fue seleccionado.

Por otra parte, se ha dicho que la información gráfica de los residuos fue de utilidad al momento de seleccionar el mejor modelo, así, basándose en el análisis gráfico de los residuos, también se seleccionó el modelo RNA4. Es decir, la FAC de errores presentó menos coeficientes de correlación que sobrepasan los límites de confianza del 95 por ciento y aunque se aprecia ligeramente significativos esto se puede considerar aceptable. Además, el periodograma acumulativo evidenció que los residuos siguen una evolución que se mantiene cercano a los intervalos de confianza establecidos, indicando que los residuos siguen una distribución muy cercana a la normal, que es lo esperable para estos modelos.

En conclusión, de lo anterior comentado, finalmente se seleccionó el modelo RNA4 como la mejor opción y más precisa para pronosticar los caudales diarios del río Amazonas a un día de anticipación. En general, este modelo logró reproducir los valores observados, siguiendo sin problemas la ligera tendencia a largo plazo y los períodos ligados a la variación estacional que presenta la serie temporal.

4.2.4. Ampliación del horizonte de pronóstico

Siendo una de las finalidades determinar un modelo, capaz de pronosticar la ocurrencia de caudales diarios con un cierto tiempo de anticipación, se optó por el modelo RNA4 como el más indicado, debido a su mejor desempeño en cotejo al resto de los modelos formulados para pronósticos a un día de anticipación. Así pues, se busca establecer su capacidad predictiva del modelo ampliando el horizonte de pronóstico (*lead time* en inglés) a múltiples días de anticipación, de tal manera, lográndose calificar sus desempeños de acuerdo a lo definido por Dawson y Wilby (2001).

Para evaluar la capacidad predictiva del modelo RNA4, se amplió el horizonte de pronóstico hasta un máximo de 30 días adelante, pero sintetizados en tres, seis, nueve, doce, quince, dieciocho, veinte uno, veinte cuatro, veinte siete y treinta días (Tabla 4). Así, resultando en la formulación de diez modelos derivados del modelo RNA4, correspondientes a cada horizonte de pronóstico, los cuales se diferencian por el patrón de salida (Tabla 4). O sea, se llevó a cabo, el pronóstico de caudales diarios del río Amazonas (estación hidrométrica de Tamshiyacu) a múltiples días de anticipación.

Tabla 4: Patrones de salida del modelo RNA4 para pronósticos de caudales a múltiples días de anticipación

Modelo	Entradas	Días adelante	Nº Entradas	Salidas
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	3 días	4	Q_{t+3}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	6 días	4	Q_{t+6}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	9 días	4	Q_{t+9}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	12 días	4	Q_{t+12}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	15 días	4	Q_{t+15}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	18 días	4	Q_{t+18}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	21 días	4	Q_{t+21}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	24 días	4	Q_{t+24}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	27 días	4	Q_{t+27}
RNA4	Q_t, Q_{t-1}, Q_{t-2} y Q_{t-3}	30 días	4	Q_{t+30}

La topología óptima de cada uno de los diez modelos formulados, se seleccionó a través de los ensayos de prueba y error, así evaluándose su desempeño alcanzado. Iniciándose con estructuras simples (una neurona oculta) e incrementándose su complejidad mediante la incorporación gradual de la cantidad de neuronas en la capa oculta. El algoritmo de

entrenamiento utilizado fue *Levenberg-Marquardt* en todos los casos, el mismo utilizado en el pronóstico a un día de anticipación.

El resumen del desempeño logrado por los nuevos modelos de RNA4 formulados para los distintos horizontes de pronóstico, se muestra en la Tabla 5, tanto para el periodo de entrenamiento, como para la validación. En ella, se resalta de manera general que los índices estadísticos de desempeño mejoran en la fase de validación, a excepción del índice MARE. Asimismo, se destaca el empeoramiento de los índices estadísticos para horizontes de pronóstico más lejanos que tres días de anticipación, de hecho, esperable.

En suma, en la fase de validación el modelo RNA4 logra alcanzar una eficiencia de *Nash-Sutcliffe* (NSE) de 0.82889 para el horizonte de pronóstico de veintiún (21) días (Tabla 5), el cual indica que el modelo es calificado como bastante bueno, de acuerdo a [Dawson y Wilby \(2001\)](#). Sin embargo, en la fase de entrenamiento el modelo logra obtener NSE = 0.77839 y RMSE = 5003.3926 m³/s, el cual indica un calificativo insatisfactorio. Por lo tanto, se seleccionó el horizonte de pronóstico anterior (18 días de anticipación), donde el modelo consigue obtener un calificativo bastante bueno con NSE = 0.82383 y RMSE = 4461.4475 m³/s (Tabla 5).

Tabla 5: Resumen de desempeño del modelo RNA4 para múltiples horizontes de pronóstico, en la fase de entrenamiento y validación

HP (días)	Entrenamiento				Validación			
	RMSE	MAE	MARE	NSE	RMSE	MAE	MARE	NSE
3	716.8912	514.9117	2.1779	0.99546	678.3206	490.7256	2.2313	0.99660
6	1629.8854	1196.6272	4.9920	0.97651	1576.5054	1160.7238	5.2309	0.98159
9	2489.6345	1862.4088	7.6839	0.94518	2364.5790	1765.3728	7.8719	0.95851
12	3247.8402	2471.3670	10.1130	0.90670	3032.6617	2299.5579	10.1630	0.93166
15	3922.4702	3045.0341	12.4360	0.86387	3669.8091	2885.8945	12.6210	0.89978
18	4461.4475	3495.7776	14.2360	0.82383	4203.7367	3344.3624	14.4820	0.86829
21	5003.3926	3964.8827	16.0830	0.77839	4787.2844	3888.1291	16.6990	0.82889
24	5430.0401	4299.4258	17.4380	0.73898	5237.2736	4278.0588	18.2990	0.79487
27	5882.4260	4708.7588	19.0990	0.69370	5656.9349	4584.0495	19.7590	0.76026
30	6193.9399	4991.2778	20.2810	0.66042	6091.8557	4977.8842	21.3140	0.72167

Nota: HP: Horizonte de pronóstico. RMSE y MAE en m³/s. MARE en por ciento (%)

Además, se evidencia que los pronósticos a veinticuatro, veintisiete y treinta días adquirieron valores de NSE de 0.79487, 0.76026 y 0.72167, respectivamente, los cuales indican

calificativos no satisfactorios, debido a que son valores inferiores a $NSE = 0.80$, según Dawson y Wilby (2001). También, se alcanzó un valor máximo de $MARE = 21.314$ por ciento, así como un valor de $RMSE = 6091.8557 \text{ m}^3/\text{s}$ para un horizonte de pronóstico de treinta días en el periodo de validación, ver la Tabla 5 y la Figura 40.

En general, el modelo RNA4 logra pronosticar hasta 21 días de anticipación los caudales observados, alcanzando un calificativo bastante bueno según el calificativo de Dawson y Wilby (2001), es decir, consigue reproducir sin problemas la tendencia decreciente que presenta la serie a lo largo de los años y los períodos ligados a la variación estacional. Además, el modelo logró conseguir un valor de $MARE = 16.699$ por ciento, $RMSE = 4787.2844 \text{ m}^3/\text{s}$ y $MAE = 3888.1291 \text{ m}^3/\text{s}$, ver la Tabla 5 y la Figura 40.

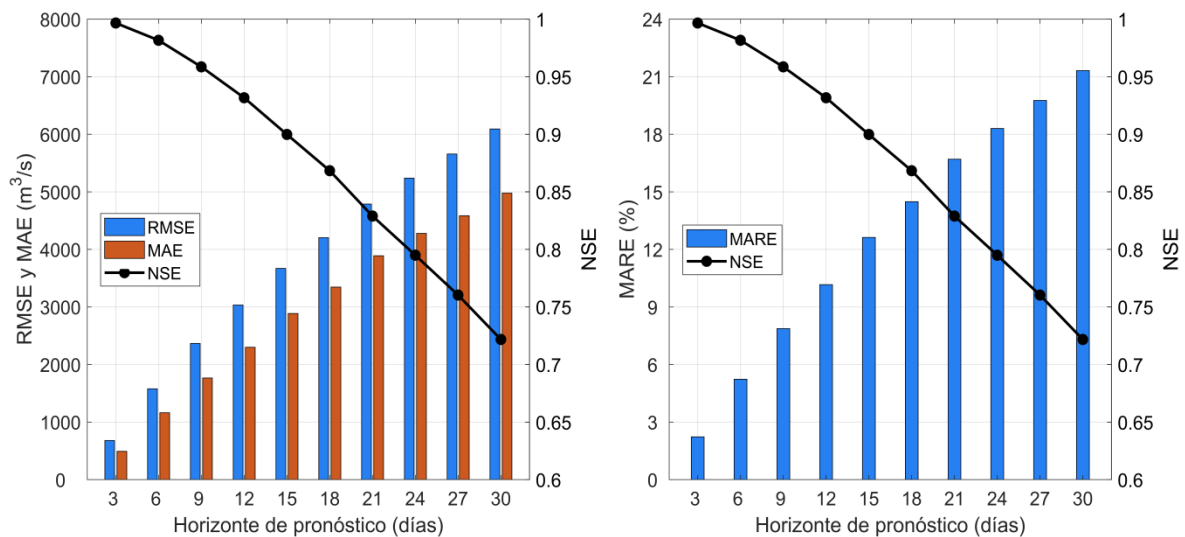


Figura 40: Desempeño del modelo RNA4 para múltiples horizontes de pronóstico en la fase de validación

En la Figura 41 (superior en azul) se muestra el hidrograma de pronósticos mediante el modelo RNA4 para 21 días de anticipación, usando datos originales del período de 01 setiembre 1985 a 31 agosto 2006, en la fase de entrenamiento de la red. Asimismo, en la misma Figura 41 (superior en rojo) se muestra el hidrograma de pronósticos también a 21 días de anticipación, usando datos de 01 setiembre 2006 hasta 31 agosto 2012, con el fin de validar la red calibrada. Por otra parte, en la misma Figura 41 (inferior izquierda) se muestra un detalle del periodo de validación, apreciándose un ligero desfase entre los valores observados y pronosticados, un fenómeno que ya ha sido reportado por otros autores, por ejemplo, Wu *et al.* (2009). Es evidente que en el pronóstico a 21 días de anticipación el modelo tiene problemas para reproducir los cambios bruscos del caudal diario. Además, se

muestra el gráfico de dispersión de los caudales observados y simulados para la fase de validación (Figura 41, inferior derecha), donde la línea roja indica la recta 1:1 que representa un pronóstico cien por ciento correcto. Los valores se mantienen en su mayoría agrupados alrededor de la línea roja, obteniéndose un coeficiente de determinación (R^2) igual a 0.8302. Es obvio, esta correspondencia debido a que el caudal del día 21 pierde la dependencia de los caudales precedentes.

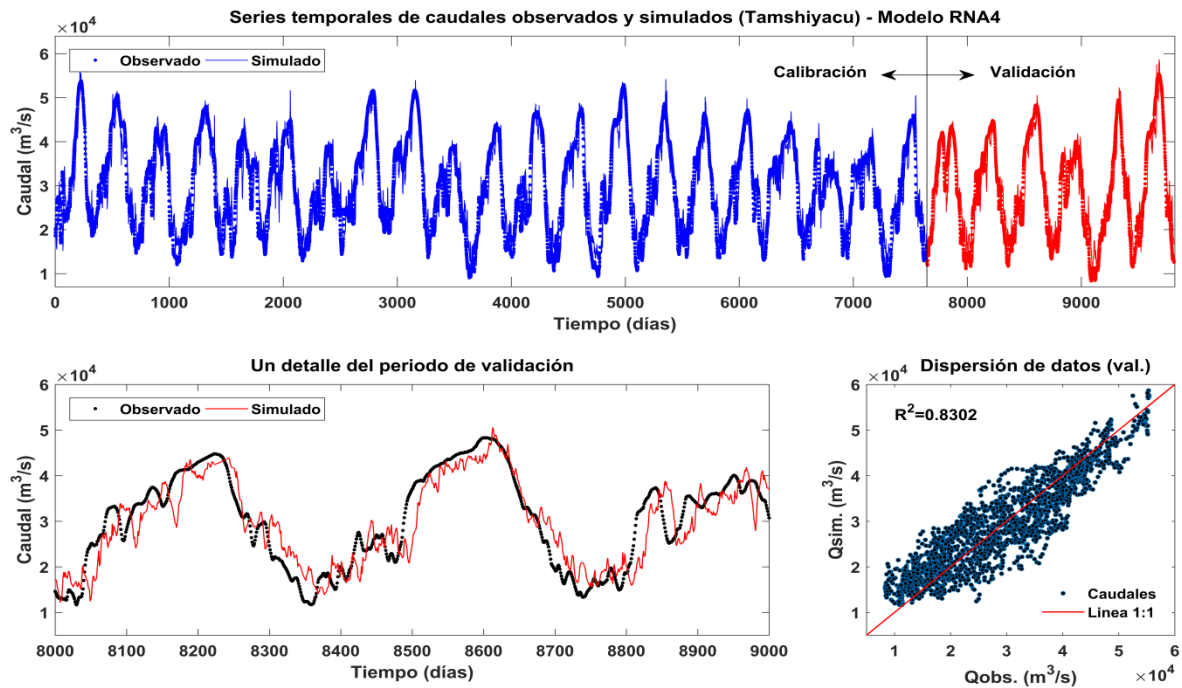


Figura 41: Hidrograma de caudales observados y pronosticados por el modelo RNA4 para 21 días de anticipación, en la fase de calibración y validación (superior), detalle y gráfico de dispersión de la fase de validación (inferior)

4.3. MODELOS HÍBRIDOS WAVELET REDES NEURONALES

En el desarrollo de este segundo grupo de modelos de la inteligencia artificial, se utilizó la red neuronal del tipo MLP de una sola capa oculta, al igual que en los modelos anteriores de RNA. Sin embargo, las entradas de datos fueron tratados de manera diferente, a partir de un análisis multiresolución utilizando la transformada *wavelet* discreta (TWD), que permite preparar las entradas a través de los coeficientes de Aproximaciones y Detalles, mediante múltiples niveles de descomposición *wavelets*, según se vio en el Capítulo III.

El criterio de segmentación de datos fue el mismo que se utilizó en los modelos de redes neuronales. Los modelos híbridos WRN se entrenaron con 7 670 muestras y se evaluaron su desempeño para la serie de datos de validación que consistió en 2 192 muestras, con el fin

de identificar el error esperado. Asimismo, la serie de datos fue escalados para que presenten valores entre $[-1, 1]$, las mismas que fueron utilizados en los modelos del primer grupo de la inteligencia artificial, es decir, los modelos de RNA.

4.3.1. Análisis de descomposición *wavelet*

El análisis de descomposición *wavelet* hace posible transmitir la información de manera separada a través de los coeficientes de Aproximaciones y Detalles. Sin embargo, para descomponer la serie original se requiere elegir una *wavelet* “madre”. Las transformadas de *wavelet Daubechies* y *Morlet* son las *wavelets* madre comúnmente usadas (Krishna *et al.* 2011). Los *Daubechies* son una familia de *wavelets* ortogonales que exhiben un buen equilibrio entre la parsimonia y la riqueza de información. En esta investigación, se eligió la función *wavelet Daubechies*, una de las familias de *wavelets* madre más ampliamente utilizados en la hidrología (Nourani *et al.* 2011b; Deka *et al.* 2012; Dadu y Deka 2013; Wei *et al.* 2013), ya que una selección de *wavelet* madre influye directamente en el tipo de Aproximación que se obtendrá. Por otro lado, también se requiere elegir el número de orden, por ende, se seleccionó el orden 2 (*db2*), debido a su buen desempeño a través del índice RMSE, logrados en la prueba realizada con el modelo RNA1 con tres niveles de descomposición *wavelet*, durante la fase de entrenamiento para las 30 primeras neuronas ocultas (Figura 42).

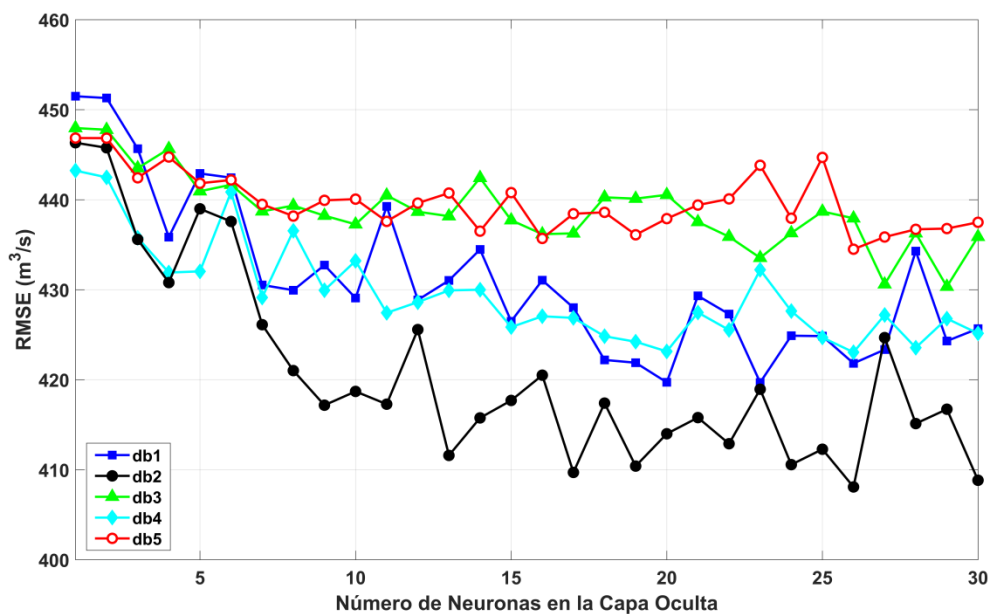


Figura 42: Selección del orden de la familia *wavelet Daubechies* en función al índice RMSE del modelo RNA1 con tres niveles de descomposición en la fase de entrenamiento, para las 30 primeras neuronas ocultas

El uso del análisis multiresolución *wavelet* implica el aumento del número de entradas a las redes neuronales en comparación a los modelos de RNA tradicionales, ya que estas entradas son los coeficientes de Aproximación y Detalles. Las Aproximaciones son los componentes de alta escala y baja frecuencia de la serie, mientras que los Detalles son los componentes de baja escala y alta frecuencia. Es importante observar que la serie original (Q) está formado por esos componentes, es decir, $Q=A_1+D_1$, donde $A_1=A_2+D_2$ o que $A_2=A_3+D_3$, y de la misma forma $Q=A_4+D_4+D_3+D_2+D_1$ (Figura 43). Este proceso de descomposición puede repetirse, con aproximaciones sucesivas que se descomponen a su vez, dividiendo en muchos componentes de resolución más baja. Esto se llama árbol de descomposición *wavelet* (Santos *et al.* 2014). Por otro lado, el número máximo de descomposiciones se determina de acuerdo con el tamaño de la serie (criterio de entropía).

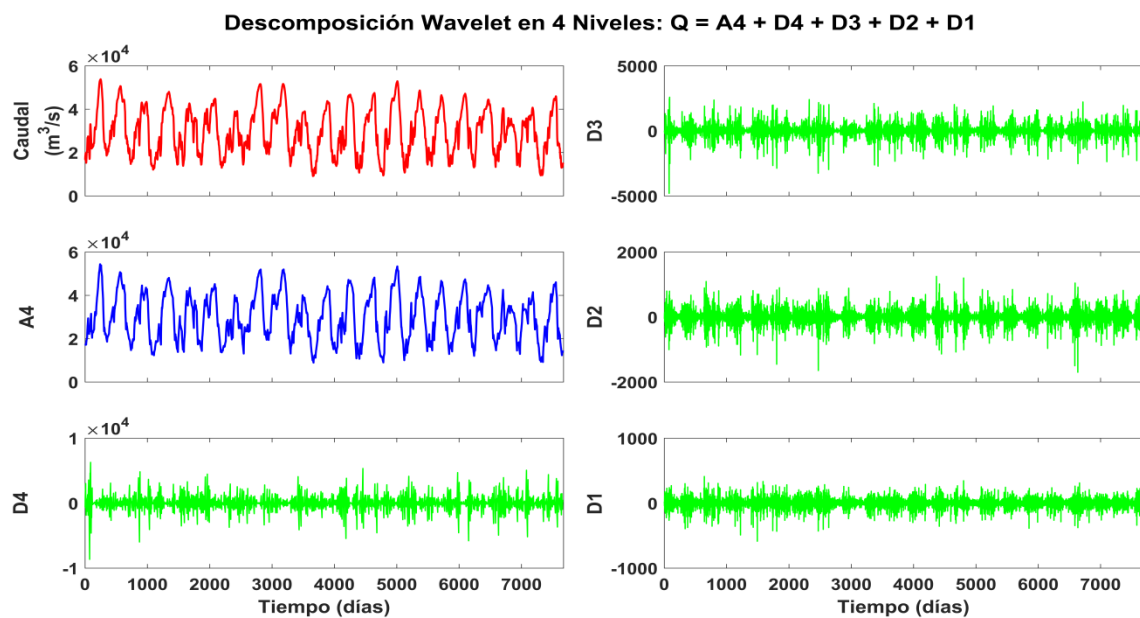


Figura 43: Descomposición *wavelet* con cuatro niveles de descomposición utilizando *wavelet* madre *Daubichies* de orden 2 (db2)

Según las directrices generales para la implementación de un modelo híbrido *wavelet* formulada en el Capítulo III de metodología, se requiere establecer el nivel óptimo de descomposición. En esta investigación, el nivel de descomposición fue determinada a partir de la metodología utilizada por Nourani *et al.* (2014a), donde el nivel de descomposición es igual a $L = ent(logN)$, siendo N el número de muestras de la serie temporal, es decir, para un total de 7 670 muestras se determinó cuatro niveles de descomposición (Figura 43). También, se examinó el efecto del segundo y tercer nivel de descomposición, aplicando a cada uno de los modelos formulados.

4.3.2. Patrones de entrada de los modelos híbridos WRN

Los patrones de entrada de los modelos híbridos WRN fueron configurados principalmente en función al máximo número de neuronas de entradas seleccionado en el primer enfoque (es decir, RNA4) y al máximo número de niveles de descomposición determinada previamente, en este caso hasta cuatro niveles de descomposición *wavelet* conforme a lo mostrado en la Figura 43. Estos patrones de entrada están representados por las señales de Aproximación y Detalles. Cabe señalar, el modelo RNA4 cuenta con un vector de cuatro entradas con valores de caudal en los instantes t , $t-1$, $t-2$ y $t-3$, los mismos que fueron considerados para los modelos híbridos WRN, así se generó cuatro grupos de combinaciones de vectores de entrada correspondientes a valores de caudal en los instantes: t ; t y $t-1$; t , $t-1$ y $t-2$; y t , $t-1$, $t-2$ y $t-3$. Asimismo, se consideró el segundo, tercer y cuarto nivel de descomposición en cada grupo (Tabla 6).

Tabla 6: Patrones de entrada y salida de los modelos híbridos WRN para pronóstico de caudales diarios a un día anticipación

Modelo	Entradas	N_d	Días adelante	N° Entradas	Salidas
WRN1	Q_t	2	1 día	3	Q_{t+1}
WRN2	Q_t	3	1 día	4	Q_{t+1}
WRN3	Q_t	4	1 día	5	Q_{t+1}
WRN4	Q_t , y Q_{t-1}	2	1 día	6	Q_{t+1}
WRN5	Q_t , y Q_{t-1}	3	1 día	8	Q_{t+1}
WRN6	Q_t , y Q_{t-1}	4	1 día	10	Q_{t+1}
WRN7	Q_t , Q_{t-1} y Q_{t-2}	2	1 día	9	Q_{t+1}
WRN8	Q_t , Q_{t-1} y Q_{t-2}	3	1 día	12	Q_{t+1}
WRN9	Q_t , Q_{t-1} y Q_{t-2}	4	1 día	15	Q_{t+1}
WRN10	Q_t , Q_{t-1} , Q_{t-2} y Q_{t-3}	2	1 día	12	Q_{t+1}
WRN11	Q_t , Q_{t-1} , Q_{t-2} y Q_{t-3}	3	1 día	16	Q_{t+1}
WRN12	Q_t , Q_{t-1} , Q_{t-2} y Q_{t-3}	4	1 día	20	Q_{t+1}

Nota: N_d es el nivel de descomposición

En la Tabla 6 se resume los patrones de entrada y salida relacionadas a combinaciones según el número de vectores de entradas y nivel de descomposición, los cuales se utilizaron para la estructuración de los modelos híbridos WRN con la finalidad de realizar pronóstico de caudales diarios a un día de anticipación. Por ejemplo, el modelo WRN12 cuenta con

el mayor número de entradas (20 neuronas), o sea, un vector de cuatro entradas y cuatro niveles de descomposición.

Los tres primeros modelos WRN1, WRN2 y WRN3 del primer grupo, corresponden a tres, cuatro y cinco vectores de entradas, respectivamente, que conciernen a los valores de caudal del día actual con dos, tres y cuatro niveles de descomposición, es decir, el modelo WRN1 cuenta con una Aproximación “A2” con baja frecuencia, y dos Detalles “D1 y D2” con alta frecuencia. El modelo WRN2 utiliza el mismo valor de caudal en el instante t , similar al modelo WRN1 con la diferencia que cuenta con tres niveles de descomposición, es decir, una Aproximación “A3” y tres Detalles “D1, D2 y D3”. El modelo WRN3 usa cuatro niveles de descomposición, es decir, una Aproximación “A4” y cuatro Detalles “D1, D2, D3 y D4” (Figura 43 y Tabla 6).

El segundo grupo de tres modelos de la Tabla 6, es decir, WRN4, WRN5 y WRN6 toman en cuenta valores de caudal en el instante t y $t-1$, y se descomponen en dos, tres y cuatro niveles de descomposición similar a los modelos precedentes, formando así, seis, ocho y diez neuronas de entrada, respectivamente. Del mismo modo, el tercer grupo correspondiente a los modelos WRN7, WRN8 y WRN9 toman en cuenta el caudal en el instante t , $t-1$ y $t-2$ con los mismos niveles de descomposición, formando así, nueve, doce y quince neuronas de entrada, respectivamente. Finalmente, el cuarto grupo correspondiente a los modelos WRN10, WRN11 y WRN12 utilizan valores de caudal en el instante t , $t-1$, $t-2$ y $t-3$ con los mismos tres niveles de descomposición.

Con la serie original de valores de caudal en el instante $t+1$ como patrón de salida (objetivo) en los doce modelos formulados. Del mismo modo, se utilizó la *wavelet* madre *Daubechies* (db2) para la descomposición de la señal original en todos los casos.

4.3.3. Modelos de WRN para pronósticos a un día

Configurado los patrones de entrada y salida de los modelos híbridos WRN, en seguida, a través del proceso de entrenamiento se determinó el número de neuronas en la capa oculta, es decir, mediante el ensayo tradicional de prueba y error. En todos los casos el algoritmo utilizado para el entrenamiento fue el de *Levenberg-Marquardt*, similar a los modelos de redes neuronales tradicionales. Luego de determinar la arquitectura óptima de los modelos híbridos WRN en la fase de entrenamiento, también, se llevó a cabo el proceso de validación utilizando la subserie de datos de caudal divididos para este fin.

En primer lugar, se estructuró el modelo WRN1 (3-N-1) que consta de una señal original (Q_t) formado por los componentes A2, D2 y D1 que representan los vectores de entrada, es decir, $Q_t = A2+D2+D1$ y una neurona de salida (Q_{t+1}) que contiene la serie de datos objetivo correspondiente para pronósticos a un día de anticipación. El desempeño de esta red óptima (es decir, 3-6-1) en la fase de entrenamiento y validación, se muestra en la primera fila de la Tabla 7. Comprobada el desempeño del modelo más simple, en seguida se desarrollaron modelos que incrementaron gradualmente el número de vectores de entrada, así como el nivel de descomposición *wavelet*, correspondientes a los modelos de WRN2 hasta WRN12.

Tabla 7: Resumen de desempeño de los modelos híbridos WRN para pronósticos a un día de anticipación, en la fase de entrenamiento y validación

Modelo híbrido	Wavelet y nivel	Estruc. RNA	Entrenamiento				Validación			
			RMSE	MAE	MARE	NSE	RMSE	MAE	MARE	NSE
WRN1	db2L2	3-6-1	426.2639	322.6163	1.2808	0.99839	407.8977	313.0631	1.3091	0.99877
WRN2	db2L3	4-13-1	411.5861	310.7090	1.2313	0.99850	406.8740	307.9250	1.3004	0.99878
WRN3	db2L4	5-29-1	404.5138	306.8899	1.2157	0.99855	400.7068	307.6678	1.2982	0.99881
WRN4	db2L2	6-15-1	110.6970	72.8986	0.3066	0.99989	107.4454	70.2281	0.3186	0.99991
WRN5	db2L3	8-17-1	100.6436	67.4412	0.2815	0.99991	99.0617	65.9224	0.2982	0.99993
WRN6	db2L4	10-17-1	97.0026	64.6810	0.2674	0.99992	96.0386	63.9161	0.2871	0.99993
WRN7	db2L2	9-33-1	93.0897	60.7858	0.2544	0.99992	89.9224	57.6599	0.2580	0.99994
WRN8	db2L3	12-30-1	87.9639	59.1919	0.2476	0.99993	87.4233	56.8572	0.2529	0.99994
WRN9	db2L4	15-26-1	77.0673	51.0971	0.2119	0.99995	80.2252	51.5704	0.2266	0.99995
WRN10	db2L2	12-36-1	73.8253	51.8713	0.2124	0.99995	74.7549	52.2975	0.2268	0.99996
WRN11	db2L3	16-28-1	70.2681	47.2861	0.1933	0.99996	71.3332	48.1056	0.2088	0.99996
WRN12	db2L4	20-23-1	67.0329	45.5260	0.1858	0.99996	67.4922	46.6375	0.2025	0.99997

Nota: db2Ln: modelo híbrido WRN de *wavelet Daubechies 2* en nivel n (2, 3 y 4); RMSE y MAE en m^3/s ; MARE en por ciento (%)

En la Tabla 7 se muestra el resumen de los criterios de desempeño (RMSE, MAE, MARE y NSE) logrados por los modelos híbridos WRN con estructuras optimizadas para pronósticos a un día de anticipación, tanto en la fase de entrenamiento o proceso de selección de la arquitectura, como en la fase de validación o prueba. En la última fila en negrita de la misma, se destaca la superioridad del modelo WRN12 (20-23-1), con respecto al resto de los modelos formulados, a través de los cuatro índices estadísticos, en ambas fases del modelamiento. Además, se resalta de manera general que para los doce redes formulados,

los tres índices RMSE, MAE y NSE mejoran en la fase de validación, sin embargo, el índice MARE empeora en esta segunda fase.

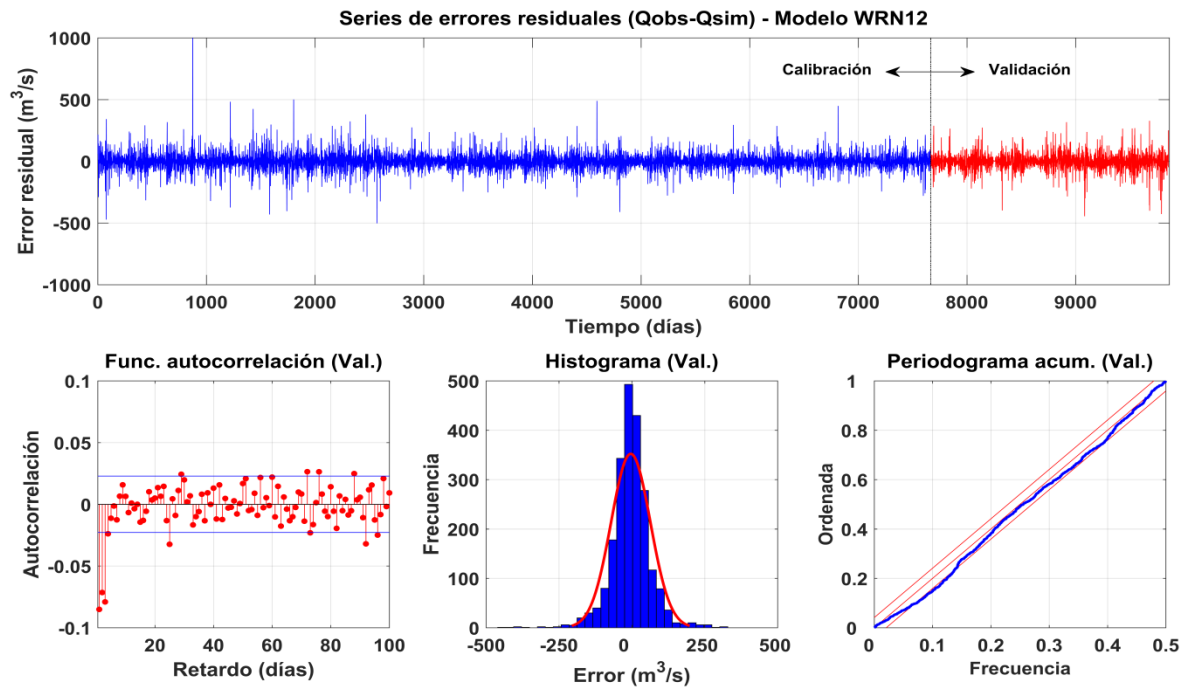


Figura 44: Serie de errores residuales del modelo WRN12 en la fase de calibración y validación (superior). Función de autocorrelación, histograma y periodograma acumulada en la fase de validación (inferior)

Además, del uso de los índices estadísticos que revelan la bondad de ajuste de los modelos híbridos WRN formulados, se evaluaron los errores residuales mediante tres criterios gráficos más comunes (función de autocorrelación, histograma y periodograma acumulada). En consecuencia, la Figura 44 parte superior, muestra la serie de errores de pronóstico del modelo híbrido WRN12 para la fase de entrenamiento y validación, los cuales parecen tener una varianza constante con magnitudes inferiores a $500 \text{ m}^3/\text{s}$ salvo en momentos puntuales. Sin embargo, el correlograma de la fase de validación (Figura 44 inferior izquierda) muestra que las autocorrelaciones para los retardos 1, 2, 3, 25, 72, 76 y 92, exceden los límites de significancia que contendrían a las correlaciones si los residuos fueran un ruido blanco. Asimismo, el periodograma acumulada de residuos (Figura 44 inferior derecha), evidencia que las sumas acumuladas del periodograma normalizadas a una escala vertical de 0 a 1, salen ligeramente de los límites de confianza del 95 por ciento que contendrían a las frecuencias si los residuos fueran aleatorios. Por último, el histograma (Figuras 44 inferior centro) muestra que la mayoría de los errores se centran

aproximadamente en cero y se distribuyen más o menos normalmente, fluctuando en el rango de $-250 \text{ m}^3/\text{s}$ hasta $250 \text{ m}^3/\text{s}$.

Según los índices estadísticos, los errores residuales logrados por el modelo híbrido WRN12 son aceptables en ambas fases del modelamiento, a pesar que el índice MARE se empeora (se incrementa en por ciento) en la fase de validación. Por otro lado, el gráfico de la FAC de los residuos (Figura 44 inferior izquierda) revela que al menos siete de los primeros 100 coeficientes de correlación estimados son ligeramente significativos para el intervalo de confianza del 95 por ciento. Por lo que los residuos conservan indicios de retardos periódicos y no son completamente un ruido blanco, como se esperaría si el modelo híbrido captara completamente la estructura de la serie. Esto se confirma si se observa el gráfico del periodograma acumulado de residuos (Figura 44 inferior derecha) donde se presenta las frecuencias acumuladas de los residuos superan mínimamente los límites críticos del 95 por ciento que contendrían a las frecuencias si los residuos fueran un ruido blanco.

a. Comparación de modelos WRN

Se han evaluado 12 modelos híbridos WRN distintos. Se diferencian entre sí tanto por la dimensión del vector de entradas, como por la cantidad de neuronas óptimas incluidas en la capa oculta. Se utilizaron vectores de entrada de dimensiones 3, 4, 5, 6, 8, 10, 9, 12, 15, 12, 16 y 20, construidos con valores de caudal actual y antecedentes, además, considerando diferentes niveles de descomposición. Los resultados logrados por cada uno de los modelos, se resumen en la Tabla 7.

La primera red híbrida WRN1 fue entrenada y evaluada su desempeño a través de cuatro estadísticos (RMSE, MAE, MARE y NSE), encontrándose una estructura optimizada de (3-6-1), que indica tres vectores de entrada, seis neuronas ocultas y un vector de salida. Los errores de pronósticos encontrados para un día de anticipación fueron: $\text{RMSE} = 407.8977 \text{ m}^3/\text{s}$, $\text{MAE} = 313.0631 \text{ m}^3/\text{s}$, $\text{MARE} = 1.3091$ por ciento y una $\text{NSE} = 0.99877$ en la fase de validación. En seguida, se incrementó el nivel de descomposición *wavelet* de tres y cuatro, para los modelos híbridos WRN2 y WRN3, reduciéndose ligeramente los errores, así obteniéndose para este último modelo valores de $\text{RMSE} = 400.7068 \text{ m}^3/\text{s}$, $\text{MAE} = 307.6678 \text{ m}^3/\text{s}$, $\text{MARE} = 1.2982$ por ciento y una $\text{NSE} = 0.99881$ en la fase de validación (Tabla 7).

Sin embargo, los modelos híbridos desde WRN4 hasta WRN11 fueron mejorando notablemente su desempeño progresivamente, en tanto se fueron agregando el número de vectores de entrada a la red a través de los niveles de descomposición *wavelet*. De este modo, consiguiéndose índices de RMSE = 107.4454 m³/s, MAE = 70.2281 m³/s, MARE = 0.3186 por ciento y una NSE = 0.99991 para el modelo WRN4; y valores de RMSE = 71.3332 m³/s, MAE = 48.1056 m³/s, MARE = 0.2088 por ciento y una NSE = 0.99996 para el modelo WRN11, ambos en el periodo de validación. Los resultados de desempeño de los doce modelos formulados para pronósticos a un día de anticipación, se detallan en la Tabla 7.

Finalmente, a partir de la Tabla 7, la arquitectura del modelo híbrido WRN12 (20-23-1) con veintitrés nodos ocultos parece tener el mejor ajuste (menor error), tanto en la fase de entrenamiento, como en la fase de validación, con RMSE = 67.0329 m³/s y 67.4922 m³/s; MAE = 45.5260 m³/s y 46.6375 m³/s; MARE = 0.1858 por ciento y 0.2025 por ciento; y NSE = 0.99996 y 0.99997, respectivamente. Por lo tanto, la arquitectura óptima seleccionada tiene 20 entradas, 23 neuronas ocultas y una salida. Esta selección es corroborada a través del análisis de los métodos gráficos de correlograma, histograma y periodograma acumulada de los errores residuales, mostrados en la Figura 44.

b. Modelo seleccionado de WRN

Se desarrollaron modelos híbridos WRN con distintas estructuras y se evaluaron sus desempeños en base a los mismos estadísticos utilizados para las redes neuronales tradicionales, cuyos resultados son especificados en las secciones precedentes. Si se seleccionara el mejor modelo en base a estos índices, ya se ha indicado que el modelo híbrido WRN12 es el que presenta mejor desempeño, tanto en la fase de entrenamiento, como en la fase de validación.

Asimismo, esta selección del modelo WRN12 es confirmada por la evaluación en base a la información gráfica de sus errores residuales, ya que su FAC presenta pocos retardos que sobrepasen los límites de confianza del 95 por ciento y no se aprecian ciclos o periodicidad en las correlaciones, estos se pueden considerar aceptable. Por otra parte, el periodograma acumulado de residuos muestra que las frecuencias se mantienen cercanas a los límites de confianza del 95 por ciento, indicando que los residuos tienen mayor probabilidad de ser aleatorios.

En resumen, se seleccionó el modelo híbrido WRN12 como la mejor opción y más precisa para pronósticos de caudales diarios del río Amazonas a un día de anticipación. En general, los resultados muestran una buena reproducción de los valores observados. Por lo tanto, este modelo seleccionado, se utilizó para prolongar a mayores horizontes de pronóstico, hasta obtener calificativos de bastante bueno, es decir, $NSE > 0.80$, según la interpretación de Dawson y Wilby (2001).

4.3.4. Ampliación del horizonte de pronóstico

Para ampliar el horizonte de pronóstico a múltiples días de anticipación, se ha identificado y seleccionado previamente el modelo que logró el mejor desempeño en el pronóstico de caudales a un día de anticipación, en este caso fue selecto el modelo WRN12. Asimismo, se evaluó su capacidad predictiva a través de los mismos criterios estadísticos utilizados en los modelos de RNA solos. De esta manera afrontar el problema de precisión que brindan los modelos de redes neuronales tradicionales.

Los patrones de entrada de estos modelos híbridos, se configuraron a partir de la misma lógica del modelo híbrido WRN12 para pronósticos a un día de anticipación, a través del análisis multiresolución *wavelet*, usando la transformada *wavelet* de familia *Daubechies* de orden 2 (db2) hasta el cuarto nivel de descomposición, así formando 20 neuronas de entradas que se mantuvieron permanentes. Con respecto a los patrones de salida, éstas consistieron de una neurona que variaron en función a cada horizonte de pronóstico, representados específicamente por los caudales en los instantes $t+3$, $t+6$, $t+9$, ... y $t+30$, los mismos horizontes considerados en los modelos de RNA solos (Tabla 8).

La Tabla 8 muestra los patrones de entrada y salida que se emplearon en los modelos híbridos *wavelet* redes neuronales (WRN12 que difieren entre sí, por el vector de salida) para múltiples horizontes de pronóstico de caudales diarios del río Amazonas (estación de Tamshiyacu, Perú).

Cabe señalar que todos los modelos formulados en esta sección utilizaron la red de tipo *perceptrón* multicapa (MLP) con tres capas: la capa de entrada; capa oculta, que utiliza una función de activación tangente sigmoideal hiperbólica; y la capa de salida, utilizando la función de activación lineal. La cantidad de neuronas de la capa oculta fueron determinadas mediante el ensayo de prueba y error, usando el algoritmo de entrenamiento *Levenberg-Marquardt*, el mismo utilizado en los modelos de RNA tradicionales.

Tabla 8: Patrones de salida del modelo híbrido WRN12 para pronósticos de caudales a múltiples días de anticipación

Modelo	Entradas	N _d	Días adelante	Nº Entradas	Salidas
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	3 días	20	Q _{t+3}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	6 días	20	Q _{t+6}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	9 días	20	Q _{t+9}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	12 días	20	Q _{t+12}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	15 días	20	Q _{t+15}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	18 días	20	Q _{t+18}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	21 días	20	Q _{t+21}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	24 días	20	Q _{t+24}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	27 días	20	Q _{t+27}
WRN12	Q _t , Q _{t-1} , Q _{t-2} y Q _{t-3}	4	30 días	20	Q _{t+30}

Nota: N_d es el nivel de descomposición *wavelet*

Se inició con la red que corresponde a un horizonte de pronóstico de tres días (Q_{t+3}), con una estructura simple, es decir, con pocas neuronas en la capa oculta, los cuales se fueron incrementando progresivamente hasta encontrar su estructura óptima. En seguida, se varió los patrones de salida en función al horizonte de pronóstico (es decir, de 6 hasta 30 días de anticipación), representados por Q_{t+6}, Q_{t+9}, Q_{t+12}, hasta Q_{t+30}. Obviamente se puede observar el crecimiento de los errores para horizontes de pronóstico más lejanos.

Tabla 9: Resumen de desempeño del modelo WRN12 para pronósticos a múltiples días de anticipación, en la fase de entrenamiento y validación

HP (días)	Entrenamiento				Validación			
	RMSE	MAE	MARE	NSE	RMSE	MAE	MARE	NSE
3	225.9347	155.3713	0.6500	0.99955	237.0071	156.5371	0.6978	0.99958
6	592.9479	412.4257	1.7323	0.99689	591.7300	416.4061	1.8831	0.99741
9	975.2460	669.2122	2.7860	0.99159	943.8651	676.9200	3.0696	0.99339
12	1409.9656	998.9213	4.1334	0.98242	1513.0280	1048.5847	4.9597	0.98299
15	2060.1585	1461.1585	6.0010	0.96245	1855.7721	1363.7556	6.0970	0.97437
18	2632.3951	1939.0622	7.9708	0.93867	2448.1966	1844.0731	8.4069	0.95533
21	3264.7400	2437.5620	9.9645	0.90565	3167.7139	2422.6458	10.9640	0.92508
24	3946.1530	3029.8809	12.3890	0.86215	3902.1614	3060.5783	13.3940	0.88613
27	4559.5595	3566.7513	14.6250	0.81597	4446.6880	3465.7110	15.1990	0.85187
30	5011.7625	3935.4421	16.0280	0.77768	4819.6705	3833.0718	16.8210	0.82578

Nota: HP: Horizonte de pronóstico; RMSE y MAE en m³/s; MARE en por ciento (%)

En la Tabla 9 se resumen los resultados de desempeño de cada uno de los diez modelos híbridos WRN formulados, que corresponden a múltiples horizontes de pronóstico, en la fase de entrenamiento y validación. Asimismo, se resalta de modo general la mejora de los estadísticos en la etapa de validación con respecto a la etapa de ajuste.

Además, en la misma Tabla, se observa que, en la fase de validación, el modelo logra obtener un calificativo bastante bueno para el horizonte de pronóstico de 30 días, según las interpretaciones de Dawson y Wilby (2001), con valores de RMSE igual a 4819.6705 m³/s, MAE igual a 3833.0718 m³/s, MARE igual a 16.821 por ciento y NSE igual a 0.82578 (última fila en negrita en la Tabla 9). Sin embargo, en la fase de entrenamiento, el modelo logró obtener un calificativo bastante bueno (NSE = 0.81597) sólo hasta el horizonte de pronósticos de 27 días (penúltima fila en negrita en la Tabla 9), con errores de RMSE igual a 4559.5595 m³/s, MAE igual a 3566.7513 m³/s y MARE igual a 14.625 por ciento. En resumen, el modelo WRN12 logra pronosticar valores de caudal observados hasta los 30 días de anticipación, así, simulando sin problema la serie histórica, la ligera tendencia decreciente y las variaciones estacionales.

La Figura 45 muestra la evolución del performance del modelo híbrido WRN12 para múltiples horizontes de pronóstico, en la fase de validación. En esta Figura se observa el crecimiento exponencial de los valores de error RMSE, MAE y MARE; asimismo, se observa un decrecimiento de modo exponencial de la eficiencia de *Nash-Sutcliffe* en los primeros 24 días de horizonte de pronóstico y un descenso lineal en los últimos tramos.

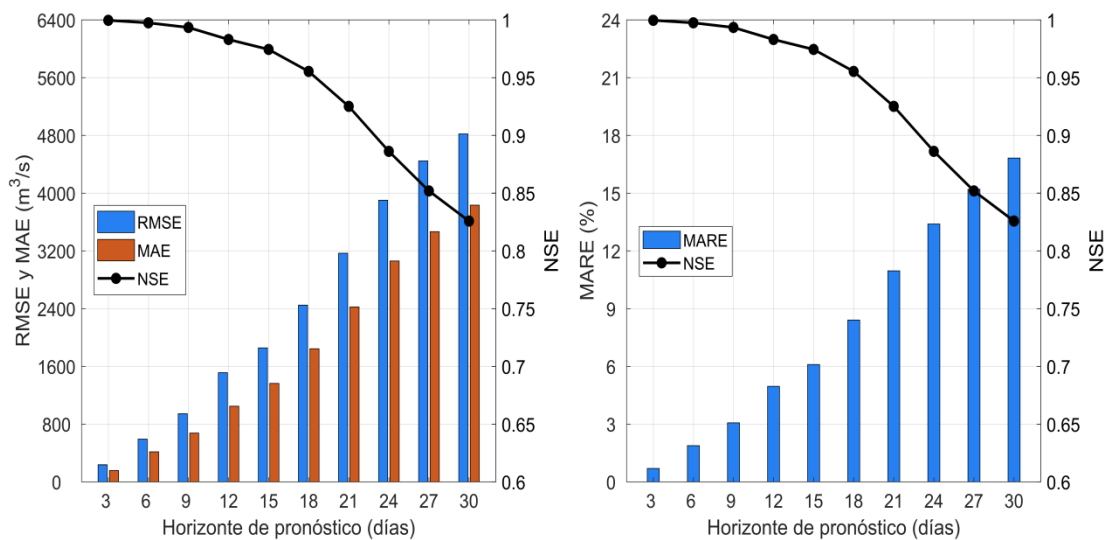


Figura 45: Desempeño del modelo híbrido WRN12 para múltiples horizontes de pronóstico en la fase de validación

La Figura 46 (superior izquierdo en azul) muestra los hidrogramas de caudales observados y pronosticados con el modelo WRN12 para pronósticos a 30 días de anticipación, en la fase de calibración, utilizando los mismos datos considerados para los modelos de RNA tradicionales. Además, en la misma Figura (superior derecho en rojo) se muestra los hidrogramas para pronósticos a 30 días de anticipación, pero para la subserie de datos de validación (seis años de registros diarios). Los errores de pronóstico obtenidos fueron RMSE igual a $5011.7625 \text{ m}^3/\text{s}$ y $4819.6705 \text{ m}^3/\text{s}$, y $\text{NSE} = 0.77768$ y 0.82578 en la fase de calibración y validación, respectivamente. También la Figura 46 (inferior izquierda) muestra un detalle del periodo de validación, donde se aprecia un ligero desfase entre los valores observados y pronosticados, un fenómeno explicado por [Wu et al. \(2009\)](#). Es evidente que, para pronósticos a 30 días de anticipación, el modelo tenga problemas para simular las variaciones del hidrograma de caudales diarios. Finalmente, se muestra el gráfico de dispersión de los caudales observados y simulados en la fase de validación (Figura 46, inferior derecha), con la recta 1:1 de color rojo que indica un pronóstico preciso. En el cual, se aprecia que los valores se mantienen en su mayoría agrupados alrededor de la línea rojo, con un $R^2 = 0.8268$. Es obvio, esta correspondencia debido a que el caudal del día 30 pierde la dependencia con los caudales antecedentes.

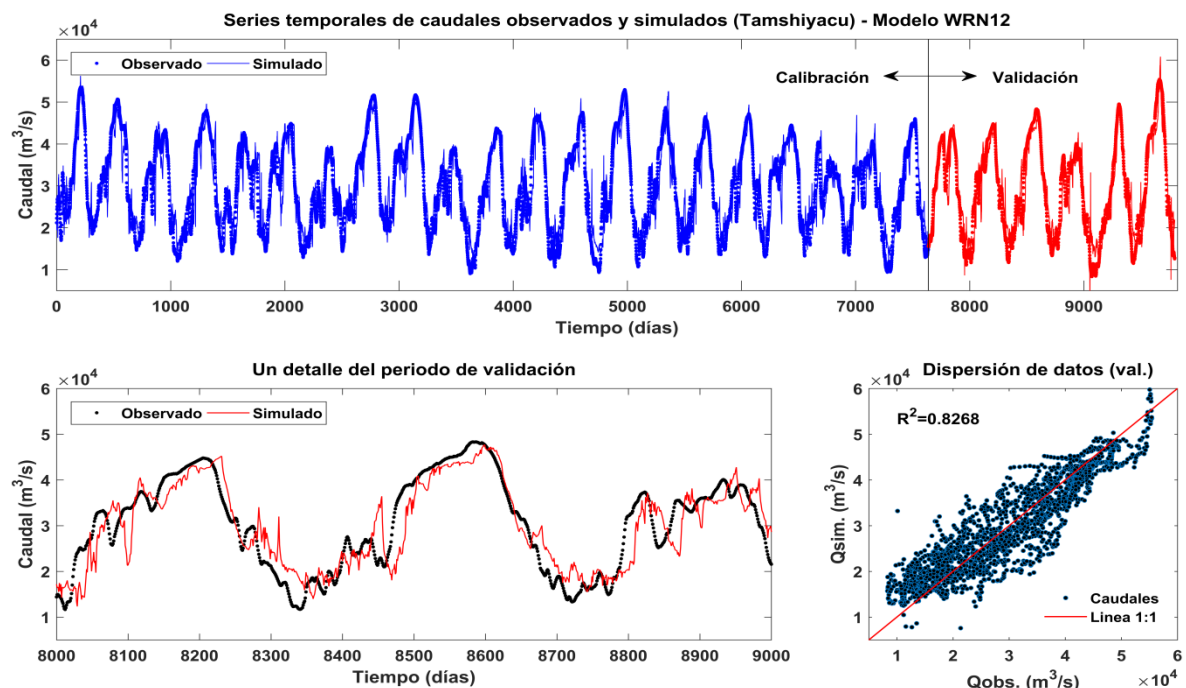


Figura 46: Hidrograma de caudales observados y pronosticados por el modelo WRN12 para 30 días de anticipación, en la fase de calibración y validación (superior), detalle y gráfico de dispersión de la fase de validación (inferior)

4.4. COMPARACIÓN DE LAS DOS TÉCNICAS DE PRONÓSTICO

De los análisis realizados en las secciones 4.2 y 4.3 de este Capítulo, queda demostrado que tanto los modelos de RNA, como los modelos híbridos WRN fueron capaces de simular el hidrograma de caudales diarios del río Amazonas. O sea, los resultados de ambas técnicas son muy aceptables, aunque difieren en la precisión, debido a que sus fundamentos parten de postulados diferentes. Por lo tanto, en esta sección se compararon ambas técnicas, en base a índices estadísticos (RMSE, MAE, MARE y NSE), a través de su mejoramiento y/o empeoramiento, en por ciento, considerando las RNA como los modelos de referencia (*Benchmark*), para pronósticos a un día de anticipación y horizontes de pronóstico más amplios.

4.4.1. Pronósticos a un día de anticipación

De las evaluaciones de pronóstico correspondiente a un día de anticipación, la mejor red formulada fue la RNA4 (4-11-1), que requiere de cuatro vectores de entrada para realizar sus pronósticos. Del mismo modo, la mejor red híbrida construida fue el modelo WRN12 (20-23-1) que requiere de la misma información original que la red de RNA4, con la diferencia que incorpora hasta cuatro niveles de descomposición *wavelet*, en otras palabras, reúne veinte vectores de entrada. En consecuencia, la comparación de técnicas se realizó entre el mejor modelo de RNA e híbridos WRN (Tabla 10, última fila en negrita), sin embargo, el cotejo se realizó también entre los modelos que incorporan menor vectores de entrada, ver la Tabla 10.

En la Tabla 10, se observa en por ciento (%) de aumento y disminución de los índices de performance (RMSE y NSE) logrados por los modelos híbridos WRN con diversas combinaciones (ver sección de formulación), en referencia a los modelos de RNA con la incorporación de la señal original sin preprocesamiento, para pronósticos a un día de anticipación, en la fase de entrenamiento y validación. En las columnas correspondientes a RMSE en por ciento (sector derecha de la Tabla 10), el signo negativo indica una mejora del índice a través de su disminución, porque cuanto más cerca de 0.0, es el valor de RMSE, es mejor. En las columnas concordantes a NSE en por ciento (sector derecha de la Tabla 10), el signo positivo muestra la mejora de este índice a través de su aumento, ya que cuanto más cerca de 1.0, es el valor de NSE, es mejor.

Tabla 10: Por ciento de aumento y disminución de los índices de desempeño del modelo híbrido WRN en comparación al modelo RNA para pronóstico a un día de anticipación, en la fase de entrenamiento y validación

Modelo RNA	Entrenamiento		Validación		Modelo Híbrido	N _d	Entrenamiento		Validación	
	RMSE	NSE	RMSE	NSE			RMSE	NSE	RMSE	NSE
RNA1	452.49	0.99819	433.28	0.99861	WRN1	2	-5.80%	0.020%	-5.86%	0.016%
					WRN2	3	-9.04%	0.031%	-6.09%	0.017%
					WRN3	4	-10.60%	0.036%	-7.52%	0.020%
RNA2	190.97	0.99968	179.32	0.99976	WRN4	2	-42.04%	0.021%	-40.08%	0.015%
					WRN5	3	-47.30%	0.023%	-44.76%	0.017%
					WRN6	4	-49.21%	0.024%	-46.44%	0.017%
RNA3	177.28	0.99972	158.36	0.99981	WRN7	2	-47.49%	0.020%	-43.22%	0.013%
					WRN8	3	-50.38%	0.021%	-44.79%	0.013%
					WRN9	4	-56.53%	0.023%	-49.34%	0.014%
RNA4	176.38	0.99972	157.67	0.99982	WRN10	2	-58.15%	0.023%	-52.59%	0.014%
					WRN11	3	-60.16%	0.024%	-54.76%	0.014%
					WRN12	4	-62.00%	0.024%	-57.19%	0.015%

Nota: N_d es el nivel de descomposición.

Es notorio que el uso del análisis multiresolución *wavelet* mejora los pronósticos en todos los casos, con resultados similares tanto en la fase de entrenamiento, como en la fase de validación (Tabla 10). O sea, del análisis comparativo entre RNA4 y WRN12, se evidencia la reducción del índice RMSE de 176.38 m³/s a 67.03 m³/s en la fase de calibración y de 157.67 m³/s a 67.49 m³/s en la fase de validación, que representa el -62.00 por ciento y -57.19 por ciento, respectivamente. En suma, el preprocesamiento de datos (es decir, la incorporación de la serie descompuesta mediante la TWD, usando *wavelet Daubechies* “db2” con diversos niveles de descomposición) mejora la precisión de pronóstico.

4.4.2. Pronósticos con horizontes más amplios

Los resultados del análisis comparativo, mostraron que si bien el modelo RNA4 logra prolongar el horizonte de pronóstico. Sin embargo, el modelo híbrido WRN12 alcanza mejorar la precisión para los mismos horizontes de pronóstico, durante la fase de entrenamiento y validación. Es decir, el método híbrido WRN puede prolongar el horizonte de pronóstico, al mismo tiempo, logra aumentar la precisión de los pronósticos.

En las Tablas 11 y 12, se aprecia en por ciento (%) de disminución y aumento de los índices de performance del modelo híbridos WRN12 en comparación al modelo RNA4, para múltiples horizontes de pronóstico, tanto en la fase de entrenamiento, como en la fase de validación, respectivamente. El signo negativo de los índices de RMSE, MAE y MARE y el signo positivo del índice NSE, todos en por ciento (sector derecha de las Tablas 11 y 12), indican una mejora de los índices a través de su disminución y aumento, respectivamente. A su vez, muestran la mejora del modelo híbridos WRN12.

Tabla 11: Comparación de desempeño de las dos técnicas de pronóstico, para múltiples horizontes de pronóstico, en la fase de entrenamiento

HP (días)	RNA4				WRN12			
	RMSE	MAE	MARE	NSE	RMSE	MAE	MARE	NSE
3	716.8912	514.9117	2.1779	0.99546	-68.48%	-69.83%	-70.16%	0.41%
6	1629.8854	1196.6272	4.9920	0.97651	-63.62%	-65.53%	-65.30%	2.09%
9	2489.6345	1862.4088	7.6839	0.94518	-60.83%	-64.07%	-63.74%	4.91%
12	3247.8402	2471.3670	10.1130	0.90670	-56.59%	-59.58%	-59.13%	8.35%
15	3922.4702	3045.0341	12.4360	0.86387	-47.48%	-52.02%	-51.74%	11.41%
18	4461.4475	3495.7776	14.2360	0.82383	-41.00%	-44.53%	-44.01%	13.94%
21	5003.3926	3964.8827	16.0830	0.77839	-34.75%	-38.52%	-38.04%	16.35%
24	5430.0401	4299.4258	17.4380	0.73898	-27.33%	-29.53%	-28.95%	16.67%
27	5882.4260	4708.7588	19.0990	0.69370	-22.49%	-24.25%	-23.43%	17.63%
30	6193.9399	4991.2778	20.2810	0.66042	-19.09%	-21.15%	-20.97%	17.76%

Nota: HP: Horizonte de pronóstico. RMSE y MAE en unidades (m^3/s).

De los resultados de las Tablas 11 y 12, se aprecia que, para horizonte de pronóstico de tres días, el modelo WRN12 presenta mejores resultados que el RNA4, pese a que, se utilizaron los mismos vectores de entrada de caudales para entrenarse y validarse. Los valores de RMSE, MAE y MARE se lograron disminuir en promedio -69 por ciento y -67 por ciento en la fase de entrenamiento y validación, respectivamente (primera fila a la derecha en las Tablas 11 y 12). Sin embargo, para horizonte de pronóstico más prolongado (30 días), los valores de RMSE, MAE y MARE se lograron disminuir en promedio -20 por ciento y -22 por ciento en la fase de entrenamiento y validación, respectivamente (última fila de la derecha en las Tablas 11 y 12). Concluyéndose que el método híbrido obtiene mejores resultados que las redes neuronales tradicionales, acorde con las evaluaciones realizadas en las comparaciones.

También, en las mismas Tablas 11 y 12, se observa que, para pronósticos a tres días de anticipación, el modelo WRN12 obtiene mejor desempeño que el modelo RNA4, evidenciándose a través del incremento del coeficiente NSE en 0.4 por ciento y 0.3 por ciento, en la fase de entrenamiento y validación, respectivamente (primera fila a la derecha en las Tablas 11 y 12). Además, para horizonte de pronóstico de 30 días, el coeficiente NSE se incrementa en 17.8 por ciento y 14.4 por ciento, durante la fase de entrenamiento y validación, respectivamente (última fila a la derecha en las Tablas 11 y 12).

Tabla 12: Comparación de desempeño de las dos técnicas de pronóstico, para múltiples horizontes de pronóstico, en la fase de validación

HP (días)	RNA4				WRN12			
	RMSE	MAE	MARE	NSE	RMSE	MAE	MARE	NSE
3	678.3206	490.7256	2.2313	0.99660	-65.06%	-68.10%	-68.72%	0.30%
6	1576.5054	1160.7238	5.2309	0.98159	-62.47%	-64.13%	-64.00%	1.61%
9	2364.5790	1765.3728	7.8719	0.95851	-60.08%	-61.66%	-61.01%	3.64%
12	3032.6617	2299.5579	10.1630	0.93166	-50.11%	-54.40%	-51.20%	5.51%
15	3669.8091	2885.8945	12.6210	0.89978	-49.43%	-52.74%	-51.69%	8.29%
18	4203.7367	3344.3624	14.4820	0.86829	-41.76%	-44.86%	-41.95%	10.02%
21	4787.2844	3888.1291	16.6990	0.82889	-33.83%	-37.69%	-34.34%	11.60%
24	5237.2736	4278.0588	18.2990	0.79487	-25.49%	-28.46%	-26.80%	11.48%
27	5656.9349	4584.0495	19.7590	0.76026	-21.39%	-24.40%	-23.08%	12.05%
30	6091.8557	4977.8842	21.3140	0.72167	-20.88%	-23.00%	-21.08%	14.43%

Nota: HP: Horizonte de pronóstico. RMSE y MAE en unidades (m^3/s).

De otro lado, las Tablas 11 y 12 evidencian que el enfoque del tipo RNA4 logra pronosticar bastante bueno hasta los 18 y 21 días de anticipación, en la fase de entrenamiento y validación, respectivamente. En contraste, el enfoque híbrido WRN consigue pronosticar con el mismo calificativo hasta los 27 y 30 días de anticipación, en las mismas fases (negrita en las Tablas 11 y 12). Es decir, el enfoque híbrido WRN posee habilidades de pronóstico nueve días más, que representan el -20.9 por ciento, -23 por ciento, -21.1 por ciento y 14.4 por ciento de mejora para los índices de RMSE, MAE, MARE y NSE, respectivamente, en la fase de validación.

La Figura 47 muestra la comparación grafica de ambas técnicas, es decir, la evolución del performance de la red neuronal RNA4 y el modelo híbrido WRN12 para diferentes horizontes de pronóstico (de 3 hasta 30 días), durante la fase de validación. Además, se

apreciar claramente el crecimiento drástico (de 678.32 m³/s hasta 6091.86 m³/s) y gradual (de 237.00 m³/s hasta 4819.67 m³/s) del índice RMSE para el modelo RNA4 y WRN12, respectivamente. Asimismo, se observa que disminuye drásticamente (de 0.9966 hasta 0.72167) y gradualmente (de 0.99958 hasta 0.82578) el índice NSE para el modelo RNA4 y WRN12, respectivamente. Sin embargo, el estadístico NSE del modelo WRN12 muestra un comportamiento lineal en el tramo final del horizonte de pronóstico.

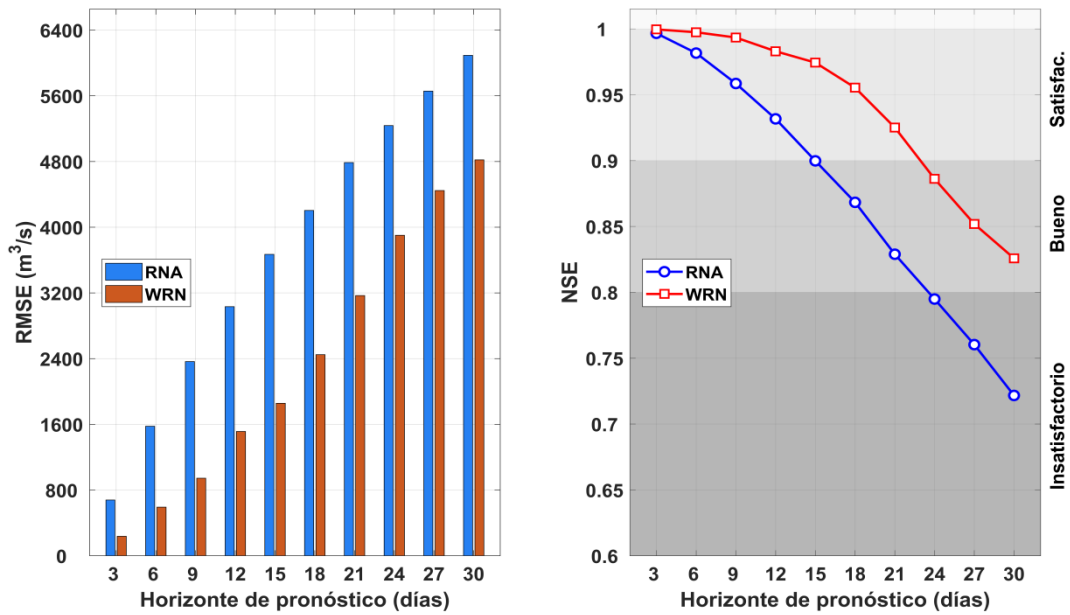


Figura 47: Comparación gráfica de performance del modelo RNA4 y el modelo híbrido WRN12 para múltiples horizontes de pronóstico en la fase de validación, a través de los estadísticos RMSE y NSE

Finalmente, se confirma que el modelo WRN12 superó significativamente al modelo RNA4 en todos los horizontes de pronóstico considerados. Reduciéndose el error RMSE en un -65.1 por ciento e incrementándose el índice NSE por encima de 0.3 por ciento para el horizonte de pronóstico de tres días. Sin embargo, el error RMSE se redujo en un -20.9 por ciento y el índice NSE se incrementó en un 14.4 por ciento para el horizonte de pronóstico más lejano (30 días), en la etapa de validación, ver la Figura 47.

V. CONCLUSIONES

- En virtud a las características altamente complejas de las series hidrológicas, se hacen difíciles los pronósticos precisos. Por ello, se han estudiado un gran número de métodos de modelamiento orientados a pronósticos, que van desde enfoques “basados en procesos” hasta “basados en datos”, para aliviar la incertidumbre. Debido a los recientes avances en las tecnologías de computación y nuevos algoritmos matemáticos, las técnicas de la IA se han convertido cada vez más comunes en el modelamiento hidrológico. Por lo tanto, en este estudio, se desarrollaron modelos “basados en datos” de pronóstico de caudales diarios con dos técnicas de la IA, la RNA y un híbrido WRN, utilizando datos históricos registrados en la estación hidrométrica de Tamshiyacu del río Amazonas.
- Se inició en el Capítulo I de introducción con la exposición de la situación problemática y la justificación de la necesidad de mejorar la calidad (tiempo de anticipación y precisión) de pronóstico a corto plazo. Ya que, efectivos modelos de pronóstico son vitales para la alerta contra inundaciones, que eviten la pérdida de vidas y materiales. Posteriormente, en los Capítulos II y III de revisión literaria y metodología, se presentaron las teorías y metodologías en las que se fundamentan las dos técnicas de pronóstico utilizadas, introduciendo conceptos y proposiciones que han permitido abordar el problema con una base metodológica sólida.
- En la primera sección del Capítulo IV de resultados, se realizó un análisis preliminar de la serie hidrológica utilizada, que revelaron información valiosa y de utilidad para la construcción de los modelos que reprodujeran adecuadamente el caudal diario. Además, en las siguientes secciones, se mostraron las resoluciones de los distintos modelos formulados, tanto las RNA solas, como los híbridos WRN para los múltiples horizontes de pronóstico, cuyos desempeños, se evaluaron, compararon y discutieron. Los resultados condujeron a encontrar el modelo más óptimo para pronosticar los caudales diarios del río Amazonas. Precizando que ambas técnicas fueron entrenadas con el 78.8 por ciento de datos, y luego validadas con el 22.2 por ciento de los datos restantes.

- Por último, se contrastaron los resultados del Capítulo IV con los objetivos que aparecen en el Capítulo I, confirmando que se logró satisfactoriamente el objetivo general de esta tesis. Es decir, se consiguió pronosticar los caudales diarios del río Amazonas, mediante un enfoque híbrido integrando la TWD y RNA, con resultados más precisos en comparación con los modelos de referencia. Además, destacar la importancia del uso de la TWD en el preprocesamiento de datos, para ampliar el horizonte de pronóstico.

Las aportaciones más relevantes que dan respuesta a los objetivos específicos del presente trabajo de tesis se mencionan a continuación:

Objetivo específico 1

En este estudio, se procuró estructurar modelos parsimoniosos. Por ende, se desarrollaron modelos univariados de RNA del tipo MLP *feedforward*. Por lo tanto, se formularon siete modelos de RNA para pronósticos a un día de anticipación (Q_{t+1}), que contienen de uno hasta siete vectores de entrada, respectivamente, considerando valores de caudal actual y antecedentes. Así, mediante el algoritmo LM, se logró entrenar y validar las relaciones entre las entradas y la salida, revelando el efecto de los caudales antecedentes en la minimización de errores. Todos los modelos formulados alcanzaron estadísticos de desempeño aceptables, sin embargo, el modelo RNA4 (4-11-1) obtuvo el mejor desempeño, cuya estructura incluye un vector de cuatro entradas (i.e., caudales del día actual y tres días antecedentes), 11 neuronas ocultas y una neurona como salida.

Objetivo específico 2

Enmarcado en pronósticos a corto plazo y a la vez selecto el modelo RNA4, se prolongaron el horizonte de pronóstico hasta los 30 días, variando el vector de salida concernientes a 3 (Q_{t+3}), 6 (Q_{t+6}), 9 (Q_{t+9}), ..., y 30 (Q_{t+30}) días de anticipación, los cuales fueron entrenados y validados con los mismos subseries usados en la primera parte. Este análisis sintetizado reveló el empeoramiento del desempeño del modelo a horizontes de pronóstico más lejanos, como fue de esperar. No obstante, permitió realizar pronósticos hasta los 15 días y 21 días de anticipación con calificativos de satisfactorio (NSE = 0.89978) y bueno (NSE = 0.82889), respectivamente, en la fase de validación. Así, estableciéndose como el modelo de referencia para la comparación con los modelos híbridos. Es evidente que la técnica de RNA es una herramienta útil y válida para realizar pronósticos a corto plazo.

Objetivo específico 3

A fin de desarrollar un enfoque híbrido, se combinaron la TWD y la RNA, utilizando la RNA del mismo tipo que, en la primera técnica, pero, con las entradas descompuestas mediante la TWD, considerando la *wavelet Daubechies* (db2) y varios niveles de descomposición. Así, se formularon 12 modelos híbridos WRN para pronósticos a un día de anticipación, cuyas entradas reúnen diversos componentes de Aproximación y Detalles. En efecto, el modelo WRN12 (20-23-1) óptimo que incluye una entrada de 20 neuronas, 23 neuronas ocultas y una neurona de salida, obtuvo pronósticos más precisos con una RMSE = 67.4922 m³/s y NSE = 0.99997. En cambio, la RNA con datos originales entregó pronósticos con una RMSE = 157.6686 m³/s y NSE = 0.99982, es decir, la RMSE se redujo en -57.19 por ciento, y la NSE aumentó en 0.015 por ciento, en la fase de validación. En suma, el modelo híbrido superó significativamente a los modelos RNA en términos de precisión.

Objetivo específico 4

Finalmente, se desarrollaron modelos WRN para múltiples días de anticipación (de 3 hasta 30 días), cuyos desempeños se cotejaron con los modelos de RNA normales, evaluándose mediante cuatro medidas estadísticas. Los resultados confirman que el modelo híbrido WRN12 superó con creces al modelo RNA4 reduciéndose la RMSE en -65.06 por ciento, mientras que la NSE se incrementó en 0.30 por ciento para el horizonte de pronóstico de tres días. Pero, la RMSE se redujo en -20.88 por ciento y la NSE se incrementó en 14.43 por ciento para pronósticos a 30 días de anticipación, en la fase de validación. Además, los resultados sintetizados, revelaron que el modelo WRN12 permitió pronosticar hasta los 21 días y 30 días de anticipación con calificativos de satisfactorio (NSE = 0.92508) y bueno (NSE = 0.82578), respectivamente, en la etapa de validación. En síntesis, los resultados mostraron la habilidad del modelo WRN para ampliar el horizonte de pronóstico.

- Esta investigación resume que la capacidad de pronóstico de los modelos RNA es mejorado cuando se adopta la técnica de TWD en el preprocesamiento de datos, confirmado a través de estadísticos de desempeño y análisis de errores. Así, mostrándose una técnica híbrida robusta y útil para mejorar la calidad de pronóstico. Por lo tanto, estos hallazgos ayudarán a los hidrólogos y tomadores de decisión en el pronóstico de caudales y la planificación y gestión sostenibles de los recursos hídricos.

VI. RECOMENDACIONES

- Evidentemente, la realidad es lo bastante compleja como para que exista una solución milagrosa que resuelva todos los problemas, en general los problemas complicados deben ser divididos en partes y cada parte ser resuelta con la técnica apropiada, ya sean convencionales, como las técnicas estadísticas, o no tan convencionales como las redes neuronales, desarrollar modelos cooperativos entre estas áreas es una tendencia en la mayoría de los trabajos actuales.
- Con base en los resultados obtenidos, como mejoras en los pronósticos de caudales diarios empleando las transformadas de *wavelets*, los trabajos futuros son incentivados para pronósticos de otras series temporales, por ejemplo, precipitación, caudales mensuales, niveles de agua subterránea, índices climáticos (por ejemplo, SPI-Índice de Precipitación Estandarizada, NDVI-Índice de Vegetación de Diferencia Normalizada, ENSO-El Niño/La Niña Oscilación del Sur, etc.) utilizando las transformadas de *wavelets*, ya sea con RNA o con otros tipos de modelos (lineales o no, autorregresivo o no).
- Otros tipos de aplicaciones también se pueden beneficiar del uso de las transformadas de *wavelets*, por ejemplo, operación de embalses, modelos de precipitación-escorrentía, entre otros. Esto para centrarse en el área de los recursos hídricos; Sin embargo, el enfoque híbrido presenta un gran potencial para otras áreas de la ingeniería (eléctrica, mecánica, producción, etc.) y economía.
- A fin de perfeccionar el estudio actual, se sugiere utilizar la estrategia actual para otras estaciones hidrométricas en la cuenca del Amazonas usando datos de caudales semanales y mensuales como diferentes características de la señal e incluyendo otras variables de series temporales como niveles de agua, lluvia, temperatura, evapotranspiración, así, generando modelos multivariados.
- Finalmente, y utilizando la analogía escuchada durante tanto tiempo de mis profesores, las RNA y modelos híbridos WRN son una herramienta más que se debe agregar a nuestro ya cargado maletín de estadísticos y técnicas de modelamiento.

VII. REFERENCIAS BIBLIOGRÁFICAS

- Abrahart, RJ; See, L. 2000. "Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments", *Hydrol. Process.* 14, 2157–2172.
- Abreu, T; Paredes, M; Araújo, KAM; Lotufo, ADP; Lopes MLM. 2015. "Electrical load forecasting using a hybrid model with artificial neural network". *Intern. J. of Comp. y Inform. Tech.*, Vol. 04, Issue 02, pp. 187-193.
- Abudu, S. 2009. "Monthly and seasonal streamflow forecasting in the Rio Grande basin". Ph.D. thesis in Civil Eng., New Mexico State University, Las Cruces, New Mexico.
- Acevedo, ML. 2009. *Computación paralela de la transformada wavelet; aplicaciones de la transformada wavelet al álgebra lineal numérica.* Tesis Doc., Valencia-España, Universidad Politécnica de Valencia, 127 p.
- Adamowski, J. 2008. "Development of a short-term river flood forecasting method for snowmelt driven floods based on wavelet and cross-wavelet analysis". *Journal of Hydrology* 353: 247–266.
- Adamowski, J; Sun, K. 2010. "Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds". *Journal of Hydrology* 390: 85–91.
- Aladag, CH; Egrioglu, E; Kadilar, C. 2009. "Forecasting nonlinear time series with a hybrid methodology". *Applied Mathematics Letters* 22, pp. 1467–1470.
- Alagha, JS; Said, MAM; Mogheir, Y. 2012. "Review – Artificial intelligence based modelling of hydrological processes", *The 4th International Engineering Conference –Towards engineering of 21st century*, pp. 13.
- Alegre, LAC. 2003. *Simulación de redes neuronales artificiales: Una aplicación didáctica.* Tesis Lic., Corrientes-Argentina, Universidad Nacional del Nordeste, pp. 188.

- Ali, MH; Abustan, I. 2014. "A new novel index for evaluating model performance". Journal of Natural Resources and Development, Vol. 55, No. unknown, pp. 1-9(9).
- Amari, SI; Murata, N; Muller, KR; Finke, M; Yang, HH. 1997. "Asymptotic statistical theory of overtraining and cross-validation". IEEE Transactions on Neural Networks 8 (5), 985–996.
- ASCE Task Committee. 2000a. "Artificial neural networks in hydrology-I: Preliminary concepts", Journal of Hydrologic Engineering, Vol. 5, No. 2, pp. 115-123.
- ASCE Task Committee. 2000b. "Artificial neural networks in hydrology-II: Hydrologic applications". Journal of Hydrologic Engineering, Vol.5, No. 2, pp.124-137.
- Aussem, A; Campbell J; Murtagh, F. 1998. "Wavelet-based feature extraction and decomposition strategies for financial forecasting". Journal of Computational Intelligence in Finance 6(2): 5-12.
- Badrzadeh, H; Sarukkalige, R. 2012. "River flow forecasting using an integrated approach of wavelet analysis and artificial neural networks". in Proceedings of the 34th Hydrology & Water Resources Symposium, Nov 19-22, pp. 1571-1578.
- Badrzadeh, H; Sarukkalige, R; Jayawardena, AW. 2013. "Impact of multi-resolution analysis of artificial intelligence models inputs on multi-step ahead river flow forecasting". Journal of Hydrology 507: 75–85.
- Badrzadeh, H. 2014. "River flow forecasting using an integrated approach of wavelet multi-resolution analysis and computational intelligence techniques". Tesis Ph.D., Curtin University, Department of Civil Engineering, pp. 191.
- Basogain, XO. 1998. Redes neuronales artificiales y sus aplicaciones. Publicaciones de la Escuela de Ingenieros, Escuela Superior de Ingeniería de Bilbao, UPV, 79 pp.
- Bates, JM; Granger, WJ. 1969. "The combination of forecasts". Operation Research Society, Vol. 20, pp. 451–468.
- Beale, MH; Hagan, MT; Demuth, HB. 2017. "*Neural Network Toolbox*", User's Guide. Versión 1. The Math Works, Inc.
- Béjar, WE; Valeriano, KY; Ilachoque, JC; Sulla, J. 2016. Predicción de caudales medios diarios en la cuenca del Amazonas aplicando redes neuronales artificiales y el modelo neurodifuso ANFIS. Research in Computing Science, 113, 23-35 pp.

- Belayneh, A; Adamowski, J. 2012. “Standard precipitation index drought forecasting using neural networks, wavelet neural networks and support vector regression”. *Journal of applied computational intelligence and soft computing*, 1-13.
- Blake, J; Francino, P; Catot, JM; Solé, I. 1995. “A comparative study for forecasting using neural networks vs genetically identified Box-Jenkins models”. *Neural Computing & Applications*, 3:139-148.
- Bowden, GJ; Maier, HR; Dandy, GC. 2002. “Optimal division of data for neural network models in water resources applications”. *Water Resources Research*, Vol. 38, 1010.
- Bowden, GJ. 2003. “Forecasting water resources variables using artificial neural networks”. Tesis de Ph.D., The University of Adelaide.
- Box, GEP; Jenkins, GM; Reinsel, G. 1976. “*Time series analysis: Forecasting and control*”. San Francisco, California: Holden-Day. Revisión de la versión de 1970.
- Bryson, AE; Ho, YC. 1969. “*Applied optimal control*”. Blaisdell, New York.
- Chatfield, C. 2001. “*Time-series forecasting*”. Chapman and Hall/CRC. pp 265.
- Clemen, R. 1989. “Combining forecasts: A review and annotated bibliography with discussion”. *International Journal of Forecasting*, 5, 559–608.
- Coifman, R; Wickerhauser, M. 1992. “Entropy-based algorithms for best basis selection”. *Information Theory, IEEE Transactions on*, 38(2):713–718.
- Coulibaly, P. 2000. “Artificial neural networks for hydrological forecasting”. Tesis Ph.D., Canada, Université Laval Québec. pp 194.
- Dadu, KS; Deka, PC. 2013. “Multistep lead time forecasting of hydrologic time series using Daubechies wavelet – Neural network hybrid model”. *Intern. Journal of Scientific & Engineering Research*, Vol. 4, Issue 10, 115-124.
- Darlane, AB; Azimi, S; Zakerinejad, A. 2014. “Artificial neural network coupled with wavelet transform for estimating snow water equivalent using passive microwave data”. *J. Earth Syst. Sci.* 123, N° 7, pp. 1591–1601.
- Dawson, CW; Wilby, RL. 2001. “Hydrological modelling using artificial neural networks”. *Progress in Physical Geography* 25, 1 pp. 80–108.

- Dawson, CW; Abrahart, RJ; See, LM. 2007. “HydroTest: A web-based toolbox of evaluation metrics for the standardised assessment of hydrological forecasts”. *Environmental Modelling & Software* 22: 1034-1052.
- Deka, PC; Prahlada, R. 2012. “Discrete wavelet neural network approach in significant wave height forecasting for multistep lead time”. *Ocean Eng.*, 43: 32-42.
- Deka, PC; Haque, L; Banhatti, AG. 2012. “Discrete wavelet-ANN approach in time series flow forecasting - A case study of Brahmaputra River”. *International Journal of Earth Sciences and Engineering*, Vol. 05, N° 04, pp. 673-685.
- DGAS (Dirección General de Aguas y Suelos, PE). 1995. *Hacia una gestión integrada de los recursos hídricos en el Perú*. Ministerio de Agricultura, Perú.
- Di, C; Yang, X; Wang, X. 2014. “A four-stage hybrid model for hydrological time series forecasting”. *PLOS ONE*, vol. 9(8): e104663.
- Dong, X; Dohmen-Janssen, CM; Booij, M; Hulscher, S. 2006. “Effect of flow forecasting quality on benefits of reservoir operation: A case study for the Geheyan reservoir (China)”. *Hydrol. Earth Syst. Sci. Discuss.*, 3, 3771–3814.
- Espinoza, CA. 2010. *Predicción a corto plazo de la demanda de agua urbana en áreas densamente pobladas*, Tesis Doc., España, Universidad Politécnica de Valencia.
- Espinoza, JC; Fraizy, P; Guyot, JL; Ordoñez, JJ; Pombosa, R; Ronchail J. 2006. “La variabilité des débits du Rio Amazonas au Pérou. Climate variability and change-hydrological impacts”. *IAHS Publ.* 308: 424-429.
- Espinoza, JC; Guyot, J; Ronchail, J; Cochonneau, G; Filizola, N; Fraizy, P; Labat, D; De Oliveira, E; Ordonez, J; Vauchel, P. 2009a. “Contrasting regional discharge evolutions in the Amazon basin (1974-2004)”. *Journal of Hydrology*, 375: 297-311.
- Espinoza, JC; Ronchail, J; Guyot, JL; Cocheneau, G; Filizola, N; Lavado, W; De Oliveira, E; Pombosa, R; Vauchel, P. 2009b. “Spatio-temporal rainfall variability in the Amazon basin countries (Brazil, Peru, Bolivia, Colombia and Ecuador)”. *International Journal of Climatology*, 29: 1574–1594.
- Espinoza, JC; Ronchail, J; Guyot, JL; Junquas, C; Vauchel, P; Lavado, W; Drapeau, G; Pombosa, R. 2011a. “Climate variability and extreme drought in the upper Solimões River (western Amazon basin): Understanding the exceptional 2010 drought”. *Geophys. Res. Lett.* 38 L13406.

- Espinoza, JC; Lengaigne, M; Ronchail, J; Janicot, S. 2011b. “Large-scale circulation patterns and related rainfall in the Amazon basin: A neuronal networks approach”. Earth and Environmental Science Climate Dynamics.
- Espinoza, JC; Ronchail, J; Guyot, JL; Junquas, C; Drapeau, G; Martinez, JM; Santini W; Vauchel, P; Lavado, W; Espinoza, R. 2012. “From drought to flooding: understanding the abrupt 2010-2011 hydrological annual cycle in the upper Solimoes River (western Amazon basin)”. Environ. Res. Lett. 7 024008.
- Espinoza, JC; Ronchail, J; Frappart, F; Lavado, W; Santini, W; Guyot, JL. 2013. “The major floods in the Amazonas River and tributaries (western Amazon basin) during the 1970-2012 period: A focus on the 2012 flood”. J. of Hydrometeorology.
- Faraway, J; Chatfield, C. 1998. “Time series forecasting with neural networks: A comparative study using the airline data”, Applied Statistics 47, pages: 231–250.
- Faruk, DO. 2010. “A hybrid neural network and ARIMA model for water quality time series prediction”, Eng. Applications of Artificial Intelligence, v. 23, p. 586-594.
- Gallardo, M. 2009. Aplicación de técnicas de clustering para la mejora del aprendizaje. Universidad Carlos III de Madrid, Leganés-España, pp. 123.
- Gestal, PM. 2013. Introducción a las redes de neuronas artificiales, Universidad de La Coruña, España, <http://sabia.tic.udc.es/mgestal/cv/RNATutorial/TutorialRNA.pdf>
- Govindaraju, RS; Rao, AR. 2000. “*Artificial neural networks in hydrology*”. Dordrech: Kluwer Academic, Water Science and Technology Library, vol. 36.
- Guyot, JL; Bazan H; Fraizy P; Ordoñez JJ; Armijos E; Laraque A. 2007. “Suspended sediment yields in the Amazon basin of Peru: A first estimation”. Water quality and sediments behaviour of the future: predictions for the 21st century, pp. 3-10.
- Haykin, S. 1998. “*Neural networks: A comprehensive foundation*”. Prentice-Hall-Tom Robbins.
- Hilera, JR; Martínez, VJ. 1995. Redes neuronales artificiales: Fundamentos, modelos y aplicaciones. Paradigma. RA-MA.
- Hingray, B; Picouet, C; Musy, A. 2015. “*Hydrology: A science for engineers*”, Taylor & Francis Group/CRC, Boca Raton, London, New York.

- Hipel, KW; McLeod, AI. 1994. “*Time series modelling of water resources and environmental systems*”, Amsterdam, Elsevier, DWS-45.
- Khashei, M; Hejazi, SR; Bijari, M. 2008. “A new hybrid artificial neural networks and fuzzy regression model for time series forecasting”. *Fuzzy Set Syst.* 159, 769-786.
- Khashei, M; Bijari, M; Ardali, GAR. 2009. “Improvement of auto-regressive integrated moving average models using fuzzy logic and artificial neural networks (ANNs)”. *Neurocomputing* 72, 956– 967.
- Khashei, M; Bijari M. 2010. “An artificial neural network (p, d, q) model for time series forecasting”. *Expert Systems with Applications* 37, 479–489.
- Khashei, M; Bijari, M; Hejazi, SR. 2012. “Combining seasonal ARIMA models with computational intelligence techniques for time series forecasting”. *Soft Computing*, Vol. 16, Issue 6, pp 1091–1105.
- Kim, T; Valdés, JB. 2003. “Nonlinear model for drought forecasting based on a conjunction of wavelet transforms and neural networks”. *Journal of Hydrologic Engineering* 8(6): 319–328.
- Kim, CK; Kwak, IS; Cha, EY; Chon, TS. 2006. “Implementation of wavelets and artificial neural networks to detection of toxic response behavior of chironomids (Chironomidae: Diptera) for water quality monitoring”. *Ecol. Modelling* 195: 61–71.
- Kingston, GB; Dandy, GC; Maier, HR. 2008. “Review of artificial intelligence techniques and their applications to hydrological modeling and water resources management. Part 1-Simulation”. In F. Columbus, editor, *Hydrology Research Trends*. Nova Science Publishers, Inc.
- Kisi, Ö. 2004. “Multi-layer perceptrons with Levenberg-Marquardt training algorithm for suspended sediment concentration prediction and estimation”. *Hydrological Sciences–Journal–des Sciences Hydrologiques*, 49(6).
- Kisi, Ö. 2008. “Stream flow forecasting using neuro-wavelet technique”. *Hydrological Processes* 22(20): 4142–4152.
- Kisi, Ö. 2009. “Neural network and wavelet conjunction model for modelling monthly level fluctuations in Turkey”. *Hydrological Processes* 23: 2081–2092.

- Kisi, Ö. 2011. "A combined generalized regression neural network wavelet model for monthly streamflow prediction". *KSCE J. of Civil Engineering*, 15(8):1469-1479.
- Kisi, Ö; Shiri, J. 2011. "Precipitation forecasting using wavelet-genetic programming and wavelet-neuro-fuzzy conjunction models". *Water Resour. Manage.* 25, 3135–3152.
- Koskela, T. 2003. "Neural network methods in analysing and modelling time varying processes". HUT, Espoo 2003 Report B35 Finland.
- Krishna, B; Satyaji Rao, YR; Nayak, PC. 2011. "Time series modeling of river flow using wavelet neural networks". *Journal of Water Resource and Protection*, 3, 50-59.
- Kumar, P; Foufoula-Georgiou, E. 1993. "A multicomponent decomposition of spatial random fields 1: Segregation of large and small-scale features using wavelet transforms". *Water Resources Research*, 29, 2515-32.
- Labat, D. 2005. "Recent advances in wavelet analyses: Part 1. A review of concepts". *Journal of Hydrology*, 314(1-4), 275-288.
- Labat, D. 2008. "Wavelet analysis of the annual discharge records of the world's largest rivers". *Adv. Water Resour.* 31, 1, 109-117.
- Laqui, WF. 2010. Aplicación de redes neuronales artificiales a la modelización y previsión de caudales medios mensuales del río Huancané. *Revista Peruana Geo-Atmosférica RPGA* (2), 30-44.
- Lavado, WS; Labat, D; Guyot, J; Ronchail, J; Ordoñez, JL. 2009. "TRMM rainfall data estimation over the Peruvian Amazon-Andes basin and its assimilation into a monthly water balance model". *New approaches to hydrological prediction in data-sparse regions*, IAHS Publ. 333.
- Lavado, WS; Ronchail, J; Labat, D; Espinoza, JC; Guyot, JL. 2012. "Basin-scale analysis of rainfall and runoff in Peru (1969–2004): Pacific, Titicaca and Amazonas watersheds". *Hydrological Sciences Journal*, 57:4, 625-642.
- Lujano, E; Lujano, A; Quispe, JP; Lujano, R. 2014. Pronóstico de caudales medios mensuales del río Ilave usando modelos de redes neuronales artificiales. *Revista de Investigaciones Altoandinas*, Vol. 16, 89-100 pp.
- Maheswaran, R; Khosa, R. 2012. "Wavelet-volterra coupled model for monthly stream flow forecasting". *Journal of Hydrol.* 450–451, 320–335.

- Maier, HR; Dandy, GC. 1996. "Neural network models for forecasting univariate time series". *Neural Network World* 6 (5), 747–771.
- Maier, HR; Dandy, GC. 1998. "The effect of internal parameters and geometry on the performance of back-propagation neural networks: An empirical study". *Environmental Modelling and Software* 13 (2), 193–209.
- Maier, HR; Dandy, GC. 2000. "Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications". *Environ. Model. Softw.*, 15(1), 101-124.
- Mallat, SG. 1989. "A theory for multi resolution signal decomposition: The wavelet representation". *IEEE Trans. Pattern Anal Mach. Intel.*, 11(7), 674–693.
- Maria, A. 1997. "Introduction to modeling and simulation". Proceedings of the 1997 winter simulation conference, State University of New York at Binghamton, NY 13902-6000, U.S.A.
- Mashor, MY. 2002. "On-line short-term streamflow forecasting using neural networks". *Friend 2002 regional hydrology: bridging the gap between research and practice*, IAHS publication 274: 8.
- Masters, T. 1993. "*Practical neural network recipes in C++*". New York: Academic Press.
- Mateo, F. 2012. *Redes neuronales y preprocesado de variables para modelos y sensores en bioingeniería*. Tesis Doc., Valencia, España, Universidad Politécnica de Valencia.
- Medeiros, MC; Veiga, A. 2000. "A hybrid linear-neural model for time series forecasting". *IEEE Transaction on Neural Networks*, 11(6), 1402–1412.
- Mishra, AK; Desai, VR. 2006. "Drought forecasting using feed-forward recursive neural network". *Ecological Modelling* 198: 127–138.
- Misiti, M; Misiti, Y; Openheim, G; Poggio, JM. 2016. "*Wavelet toolbox*". User's Guide. Versión 2. The Math Works, Inc.
- Molinier, M; Guyot, JL; Oliveira, E; Guimarães, V. 1996. "Les régimes hydrologiques de l'Amazone et de ses affluents. L'hydrologie tropicale: géoscience et outil pour le développement". Paris, Mai 1995. IAHS Publ. 238: 209-222.
- Montaño, JJ. 2002. *Redes neuronales artificiales aplicadas al análisis de datos*. Tesis Doc., Palma de Mallorca, España, Universidad de las Islas Baleares.

- Montgomery, DC; Jennings, CL; Kulahci, M. 2015. *“Introduction to time series analysis and forecasting”*. John Wiley & Sons, 2da edic. New Jersey.
- Morales, IO. 2010. Comparación teórico práctica entre modelos estadísticos y el *perceptrón* multicapa. Tesis Lic., Pontificia Universidad Católica Valparaíso, 110 p.
- Nash, JE; Sutcliffe, JV. 1970. “River flow forecasting through conceptual models part I – A discussion of principles”. *Journal of Hydrology*, 10(3) 282–290.
- Nemec, J. 1986. *“Hydrological Forecasting: Design and operation of hydrological forecasting systems”*. Water science and technology library, vol. 5, D. Reidel Publishing Company.
- Nourani, V; Alami, MT; Aminfar, MH. 2009a. “A combined neural-wavelet model for prediction of Ligvanchai watershed precipitation”. *Eng. Appl. Art. Int.* 22:466-472.
- Nourani, V; Komasi, M; Mano, A. 2009b. “A multivariate ANN-wavelet approach for rainfall–runoff modeling”. *Water Resour. Manage.* 23(14), 2877–2894.
- Nourani, V; Kisi, Ö; Komasi, M. 2011a. “Two hybrid artificial intelligence approaches for modeling rainfall–runoff process”. *J. Hydrol.* 402(1), 41–59.
- Nourani, V; Komasi, M; Alami, MT. 2011b. “Hybrid wavelet–genetic programming approach to optimize ANN modeling of rainfall–runoff process”. *Journal Hydrol. Eng.* 17(6), 724–741.
- Nourani, V; Baghanam, HA; Adamowski, J; Gebremichael, M. 2013. “Using self-organizing maps and wavelet transforms for space-time pre-processing of satellite precipitation and runoff data in neural network based rainfall-runoff modeling”. *J. Hydrol.* 476, 228–243.
- Nourani, V; Baghanam, HA; Adamowski, J; Kisi, Ö. 2014a. “Applications of hybrid wavelet–artificial intelligence models in hydrology: A review”. *Journal of Hydrology* 514, 358-377.
- Nourani, V; Khanghah, TR; Baghanam AH. 2014b. “Implication of feature extraction methods to improve performance of hybrid wavelet-ANN rainfall-runoff model. *Case Studies in Intelligent Computing*, pp. 457–498.
- Orr, MJL. 1996. “Introduction to radial basis function networks”. Centre for Cognitive Science, University of Edinburgh, Scotland.

- Pai, PF; Lin, CS. 2005. "A hybrid ARIMA and support vector machines model in stock price forecasting". *Omega*, v. 33, p. 497-505.
- Partal, T; Kisi, Ö. 2007. "Wavelet and neuro-fuzzy conjunction model for precipitation forecasting". *Journal of Hydrology* 342: 199–212.
- Partal, T; Cigizoglu, HK. 2008. "Estimation and forecasting of daily suspended sediment data using wavelet–neural networks". *Journal of Hydrology* 358: 317–331.
- Partal, T. 2009. "Modelling evapotranspiration using discrete wavelet transform and neural networks". *Hydrological Processes* 23: 3545–3555.
- Pujol, LR. 2009. Predicción de caudales en tiempo real en grandes cuencas utilizando redes neuronales artificiales. Tesis Doc., Ingeniería Hidráulica y Medio Ambiente, Universidad Politécnica de Valencia, España, pp-288.
- Qi, M; Zhang, PG. 2001. "An investigation of model selection criteria for neural network time series forecasting". *European J. Operational Research*. 132, p.666-680.
- Rivera, E. 2005. Introducción a redes neuronales artificiales. Universidad Don Bosco – Científica 6.
- Ronchail, J; Espinoza, JC; Drapeau, G; Sabot, M; Cochonneau, G; Schor, T. 2018. "The flood recession period in western Amazonia and its variability during the 1985–2015 period", *Journal of Hydrology: Regional Studies* 15, 16–30.
- Rumelhart, DE; Hinton, GE; Williams, RJ. 1986. "Learning internal representations by error propagation", in Rumelhart, DE and McClelland, JL, Eds., *Parallel Data Processing*, Vol. 1, Cambridge, MA: MIT Press, pp. 318-362.
- Salinas, MD. 2009. Diseño e implementación de un sistema de control basado en redes neuronales para el proceso de destilación de alcoholes. Tesis de Maestría, Universidad Nacional de Ingeniería, Lima-Perú.
- Sánchez, PA; Velásquez, JD. 2010. Problemas de investigación en la predicción de series de tiempo con redes neuronales artificiales. *Revista Avances en Sistemas e Informática*. 7(3), pp.67-73.
- Sánchez, PA. 2012. Una nueva metodología de entrenamiento de redes neuronales y sus implicaciones en la selección de modelos. Tesis Doc., Universidad Nacional de Colombia, pp.125.

- Sánchez, A. 2014. Máquinas de aprendizaje extremo multicapa: Estudio y evaluación de resultados en la segmentación automática de carótidas en imágenes ecográficas. Tesis Pregrado, Universidad Politécnica de Cartagena, España, pp. 94.
- Sang, YF. 2012. “A practical guide to discrete wavelet decomposition of hydrologic time series”. *Water Resources Management*, 26:3345–3365.
- Santos, CAG; Freire, PKMML; Silva, GBL; Silva, RM. 2014. “Discrete wavelet transform coupled with ANN for daily discharge forecasting into Três Marias reservoir”. *Int. Assoc. of Hydr. Sc.* 364, Bologna, Italy, pp. 100-105.
- Santos, CAG; Silva, GBL. 2013. “Daily streamflow forecasting using a wavelet transform and artificial neural network hybrid models”. *Hydrol. Sci. J.* 59(2), 321–324.
- Shabri, A; Samsudin, R. 2014. “Daily crude oil price forecasting using hybridizing wavelet and artificial neural network model”. *Mathematical Problems in Engineering*, ID 201402, 10 pp.
- Shafaei, M; Kisi, Ö. 2016. “Predicting river daily flow using wavelet-artificial neural networks based on regression analyses in comparison with artificial neural networks and support vector machine models”. *Neural Comput. & Applic.* 28, 15–28.
- Shamseldin, AY. 2004. “Hybrid neural network modelling solutions”. in RJ. Abrahart, PE. Kneale, and LM. See (Eds.). *Neural networks for hydrological modeling*. A.A. Balkema Publishers, Leiden, 61-79.
- Shoaib, M; Shamseldin, AY; Melville, BW. 2014a. “Comparative study of different wavelet based neural network models for rainfall-runoff modeling”. *Journal Hydrology* 515, 47–58.
- Shoaib, M; Shamseldin, AY; Melville, BW; Khan, MM. 2014b. “Hybrid wavelet neuro-fuzzy approach for rainfall-runoff modeling”, *J. Comput. Civil Eng.*
- Shoaib, M; Shamseldin, AY; Melville, BW; Khan, MM. 2015. “Runoff forecasting using hybrid wavelet gene expression programming (WGEP) approach”. *J. Hydrol.* 527, 326–344.
- Shoaib, M; Shamseldin, AY; Melville, BW; Khan, MM. 2016. “Hybrid wavelet neural network approach”. *Artificial Neural Network Modelling* Vol. 628 of the series *Studies in Computational Intelligence*, pp. 127-143.

- Singh, RM. 2011. "Wavelet-ANN model for flood events". Proceedings of the international conference on soft computing for problem solving, pp. 165-175.
- Sivakumar, B; Berndtsson, R. 2010. "*Advances in data-based approaches for hydrologic modeling and forecasting*". World Scientific, pp. 519.
- Socha, DF; Ortiz, GA. 2005. Aplicación de redes neuronales MLP a la predicción de un paso en series de tiempo. Fundación Universitaria Konrad Lorenz, Bogotá, pp. 183.
- Talento, S. 2011. Bases para un sistema de predicción de caudales de aporte a Rincón del Bonete y Salto Grande. Tesis de Maestría, Montevideo, Universidad de la República de Uruguay.
- Tang, Z; Fishwick, PA. 1993. "Feedforward neural nets as models for time series forecasting". ORSA Journal on Computing. 5(4), p.374–385.
- Tukey, JW. 1977. Exploratory data analysis. New York: Addison-Wesley.
- Van den Boogaard, HFP; Gautam, DK; Mynett, AE. 1998. "Auto-regressive neural networks for the modelling of time series". In: Hydrodynamics 98, Babovic and Larsen (Eds.), Balkema, Rotterdam, 741–748.
- Veitch, D. 2005. "Wavelet neural networks and their application in the study of dynamical systems". Tesis M.Sc. in Data Analysis, Networks and Nonlinear Dynamics, University of York, UK.
- Viessman, W; Lewis, GL; Knapp, JW. 1989. "*Introduction to hydrology*". Harpercollins College Div., Fourth Edition, 751 pp.
- Voort, MVD; Dougherty, M; Watson, S. 1996. "Combining Kohonen maps with ARIMA time series models to forecast traffic flow". Transportation Research Part C: Emerging Technologies, 4, 307–318.
- Wanas, N; Auda, G; Kamel, MS; Karray, F. 1998. "On the optimal number of hidden nodes in a neural network". In: Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering. Michigan. IEEE p. 918-921.
- Wang, W. 2006. "*Stochasticity, nonlinearity and forecasting of streamflow processes*", IOS Press: Amsterdam, The Netherlands.

- Wang, WC; Chau, KW; Cheng, CT; Qui, L. 2009. "A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series". *J. Hydrol.* 374, 294-306.
- Wang, W; Ding, J. 2003. "Wavelet network model and its application to the prediction of hydrology". *Nature and Science*, Vol. 1, No. 1, pp. 67-71.
- Wedding, DK; Cios, KJ. 1996. "Time series forecasting by combining networks, certainty factors, RBF and the Box-Jenkins model". *Neurocomputing*, 10, 149-168.
- Wei, S; Song, J; Khan, NI. 2012. "Simulating and predicting river discharge time series using a wavelet-neural network hybrid modelling approach". *Hydrological Processes*, 26, 281-296.
- Wei, S; Yang, H; Song, JX; Abbaspour, K; Xu, ZX. 2013. "A wavelet-neural network hybrid modelling approach for estimating and predicting river monthly flows". *Hydrological Sciences Journal*, 58 (2), 374-389.
- Werbos, P. 1974. "Beyond regression: New tools for predicting and analysis in the behavioral sciences". Tesis Ph.D., Cambridge, MA: Harvard University Committee on Applied Mathematics.
- WMO168, World Meteorological Organization. 2009. "Guide to hydrological practices: Volume II, management of water resources and application of hydrological practices". WMO N°168, Sixth edition.
- Wu, CL; Chau, KW; Li, YS. 2009. "Methods to improve neural network performance in daily flows prediction". *Journal of Hydrology* 372: 80-93.
- Yan, H; Zou, Z. 2013. "Application of a hybrid ARIMA and neural network model to water quality times series forecasting". *J. Converg. Inf. Technol.*, vol. 8(4).
- Yu, L; Wang, S; Lai, KK. 2009. "A neural-network-based nonlinear metamodeling approach to financial time series forecasting". *Applied Soft Comp.* 9, p.563-574.
- Yu, H; Wilamowski, BM. 2010. "Levenberg-Marquardt training". Auburn University. Enlace: http://www.eng.auburn.edu/~wilambm/pap/2011/K10149_C012.pdf
- Zealand, CM. 1997. "Short term streamflow forecasting using artificial neural networks". Tesis de M.Sc., The University of Manitoba, Canada.

- Zhang, G; Patuwo, EB; Hu, MY. 1998. "Forecasting with artificial neural networks: The state of the art". *International Journal of Forecasting* 14, 35-62.
- Zhang, GP; Patuwo, BE; Hu, MY. 2001. "A simulation study of artificial neural networks for nonlinear time-series forecasting". *Comput. & Oper. Res.*, 28(4), pp.381-396.
- Zhang, GP. 2003. "Time series forecasting using a hybrid ARIMA and neural network model". *Neurocomputing* 50, 159-175.
- Zhang, GP. 2007. "A neural network ensemble method with jittered training data for time series forecasting". *Information Sciences*, vol. 177, no. 23, pp.5329-5346.
- Zubieta, R; Geritana, A; Espinoza, JC; Lavado W. 2015. "Impacts of satellite-based precipitation datasets on rainfall-runoff modeling of the western Amazon basin of Peru and Ecuador". *Journal Hydrology*, 528, 599-612.
- Zubieta, R; Getirana, A; Espinoza, JC; Lavado, W; Aragon, L. 2017. "Hydrological modeling of the Peruvian-Ecuadorian Amazon basin using GPM-IMERG satellite-based precipitation dataset". *Hydrol. Earth Syst. Sci.*, 21, 3543-3555.

VIII. ANEXOS

ANEXO 1: Generalidades del *software Matlab*

Matlab es un programa interactivo para computación numérica y visualización de datos. O sea, integra análisis numérico, cálculo matricial, proceso de señal y representación gráfica en un entorno completo donde los problemas y sus soluciones son expresados sin necesidad de escribir un programa. Además, posee una extraordinaria versatilidad y capacidad para resolver complicados problemas en matemática aplicada, física, química, ingeniería, finanzas y muchas otras aplicaciones. Está basado en un sofisticado *software* de matrices para el análisis de sistemas de ecuaciones.

Matlab es un potente lenguaje diseñado para la computación técnica. El nombre *Matlab* proviene de la contracción de los términos *MATRIX LABORATORY*, dado que el tipo de dato básico que gestiona es una matriz (*array*) y fue inicialmente concebido para proporcionar fácil acceso a las librerías *Linpack* y *Eispack*, las cuales representan hoy en día dos de las librerías más importantes en computación y cálculo matricial.

Matlab se ha convertido en una herramienta básica, tanto para los profesionales y centros de investigadores, como una importante herramienta para la impartición de cursos universitarios, tales como sistemas e ingeniería de control, álgebra lineal, proceso digital de imagen, señal, modelado y simulación, etc. En el mundo industrial, *Matlab* está siendo utilizado como herramienta de investigación para la resolución de complejos problemas planteados en la realización y aplicación de modelos matemáticos en ingeniería.

Matlab dispone también en la actualidad de un amplio abanico de programas de apoyos especializados, denominados *Toolboxes*, que extienden significativamente el número de funciones incorporadas en el programa principal. Estos *Toolboxes* cubren en la actualidad prácticamente casi todas las áreas principales en el mundo de la ingeniería y la simulación, destacando entre ellos el '*toolbox*' de proceso de imágenes, señal, control robusto, estadística, análisis financiero, matemáticas simbólicas, redes neurales, lógica difusa, identificación de sistemas, simulación de sistemas dinámicos, *wavelets*, etc.

ANEXO 2. Metodología de redes neuronales en *Matlab*

La metodología a emplear para la resolución de un problema mediante RNA es, en lo que a aspectos generales se refiere, independiente de la naturaleza del problema en cuestión. Sin embargo, existen particularidades, puntos clave en los que unos problemas difieren de otros como es, por ejemplo, el algoritmo de aprendizaje empleado. En este apartado se trata de presentar dichas similitudes y diferencias y se describen los pasos a seguir para resolver el problema. La *Toolbox* “*Neural Network*” de *Matlab* proporciona funciones para el análisis de un gran número de redes neuronales artificiales, para mayor detalle revisar la guía de usuario de [Beale et al. \(2017\)](#).

2.1. Recoger y preparar los datos

El objetivo de esta primera etapa es, en base al conjunto de variables disponibles y a los resultados que se quiere obtener, hacer una primera selección de aquellas variables que se considera de mayor importancia en la salida de nuestra red. Esta decisión es importante en la medida en que un número alto de variables podría ayudar a obtener mejores predicciones en lo que al encuentro de nuestros patrones se refiere (si es esa, por ejemplo, la aplicación que se le desea dar) pero puede suponer también un costo computacional alto. El proceso de optimización de tiempo y fiabilidad de este tipo de herramientas se basa en una decisión inicial fundamentada en una serie de restricciones y cuestiones: ¿Qué error estoy dispuesto a asumir? ¿Hasta qué punto el tiempo es un factor crítico? La respuesta a estas preguntas supone el punto de partida de un proceso de realimentación de tal manera que mediante la observación de los resultados se habrá de ver si es interesante o no aumentar el número de variables o incluso reducirlo, buscando ahorrar tiempo a cambio de una reducción controlada de la fiabilidad.

Es importante destacar que las bases de datos utilizadas en general en problemas de redes neuronales son bastante amplias. Ello implica la necesidad de llevar a cabo un buen análisis de los mismos para evitar errores asociados a aspectos como campos sin rellenar, cifras incoherentes, mala colocación de comas o puntos decimales, etc. Se genera así una matriz “*input*” con muestras y variables.

Tras esta preparación es necesario que los valores numéricos de variables para cada uno de las muestras sean introducidos mediante algún tipo de función dentro de un intervalo teniendo en cuenta la función de activación que se usará en la neurona. Ello lo que permite

es que la consideración de la influencia de una determinada variable o característica se haga de manera proporcional y equitativa, independientemente del valor numérico que lleve asociado. En principio, la función de activación que se usará será la tangente sigmoidea y lo que esta función requiere es que se introduzca en ella valores situados en el intervalo [-1,1]. La función que implementa por defecto es precisamente “*mapminmax*” suministrado por el Toolbox “*Neural Network*” de *Matlab*, que acota en [-1,1] los valores numéricos de los “*inputs*”.

2.2. División de los datos en tres grupos

De todas las muestras que se tenga habrá de seleccionar tres grupos:

- Grupo de entrenamiento: Supone el 80 por ciento del total de las muestras disponibles. Se utiliza para entrenar la red, es decir, permiten que los pesos varíen en función de sus características y valores.
- Grupo de validación: Supone el 10 por ciento del total de las muestras disponibles. Se utiliza para ajustar ciertos valores de parámetros asociados al entrenamiento.
- Grupo de *test*: Supone el 10 por ciento del total de las muestras disponibles. Se utiliza para comprobar la fiabilidad de la red creada.

Los por cientos señalados anteriormente son orientativos y pueden ser variados en caso de considerarse necesario. Por lo común, también se suele utilizar un 70 por ciento de los datos para entrenar la red, un 15 por ciento para validación, y el 15 por ciento restante para *test*.

Para la selección de las muestras que pertenecerán a cada uno de los grupos se usará una función *random* que selecciona aleatoriamente de acuerdo con los por cientos marcados.

2.3. Creación, configuración y entrenamiento de la red

Existen múltiples maneras de llevar a cabo este proceso. Para el caso de redes multicapa entrenadas por retropropagación una forma sería introducir el siguiente código:

```
net = feedforwardnet;  
net = configure(net, Inputs, Targets);  
net = init(net);
```

El primer comando crea la red, el segundo la configura otorgándole a los parámetros valores por defecto que se puede modificar y establece aleatoriamente unos valores para los pesos y los sesgos dentro de un rango fijando, y el tercer comando reinicializa los valores asociados a pesos y sesgos.

Sin embargo, teniendo clara la naturaleza del problema a resolver, se puede condensar estos tres comandos en uno solo con el que, además, se define la arquitectura de la red y una serie de funciones y parámetros que, a priori, optimizan el proceso. Los problemas que se abordarán a lo largo de este trabajo serán o bien de ajuste de funciones o bien de clasificación.

Para el problema de clasificación se usa el siguiente comando:

```
net = patternnet(hiddenLayerSize, 'trainscg');
```

donde *hiddenLayerSize* es la variable que almacena el número de neuronas de la capa oculta mediante un escalar si es única o en forma de vector si es múltiple y *trainscg* es la función de entrenamiento.

Para el problema de ajuste de funciones se usa el siguiente comando:

```
net = fitnet(hiddenLayerSize);
```

donde *hiddenLayerSize* representa lo mismo que para el caso de clasificación. En el problema de ajuste no es necesario seleccionar una función de entrenamiento concreta pues *Matlab* toma, por defecto, *trainlm*.

El entrenamiento que se establece por defecto es el *batch*. De acuerdo con la guía del *Tooblox* de *Matlab*, es más rápido y genera menos errores que el entrenamiento incremental.

Dependiendo de la aplicación, será mejor utilizar una función u otra. La función de entrenamiento más rápida es *trainlm*. *Trainbfg* también es bastante rápida. Sin embargo, estas funciones son menos eficientes cuanto más grandes son las redes porque necesitan mucha memoria. Funciones mucho más eficientes son *trainscg* o *trainrp*, que no gastan mucha memoria y son más rápidas que el gradiente descendente. En principio, tal y como ya se ha señalado, se usa *trainlm* para ajuste de funciones y *trainscg* para clasificación.

Como también se ha señalado anteriormente la técnica que se usa para el entrenamiento de la red es la retropropagación, dentro de la cual hay distintos algoritmos de entrenamiento:

Función	Descripción
trainb	Entrenamiento <i>batch</i> con reglas de entrenamiento para pesos y umbrales
trainbfg	<i>BFGS quasi-Newton backpropagation</i>
trainbfgc	<i>BFGS quasi-Newton backpropagation</i> para controladores adaptativos de referencia de modelos RNA
trainbr	Regularización bayesiana
trainbuwb	Entrenamiento <i>batch</i> no supervisado para pesos y umbrales
trainc	Actualización incremental de orden cíclico
traincgb	<i>Powell-Beale backpropagation</i> de gradiente conjugado
traincgf	<i>Fletcher-Powell backpropagation</i> de gradiente conjugado
traincgp	<i>Polak-Ribière backpropagation</i> de gradiente conjugado
traingd	<i>Backpropagation</i> por descenso de gradiente
traingda	<i>Backpropagation</i> por descenso de gradiente con reglas de entrenamiento adaptativas
traingdm	<i>Backpropagation</i> por descenso de gradiente con momento
traingdx	<i>Backpropagation</i> por descenso de gradiente con momento y reglas de entrenamiento adaptativas
trainlm	<i>Levenberg-Marquardt backpropagation</i>
trainoss	<i>One step secant backpropagation</i>
trainr	Entrenamiento incremental de orden aleatorio con funciones de aprendizaje
trainrp	<i>Resilient backpropagation</i> (Rprop)
trains	Entrenamiento incremental de orden secuencial con funciones de aprendizaje
trainscg	<i>Backpropagation</i> por descenso de gradiente conjugado y escalado

2.4. Algoritmo de entrenamiento

El algoritmo *Levenberg-Marquardt* proporciona una solución numérica al problema de minimizar una función no lineal. Es rápido y proporciona una convergencia estable (Yu y Wilamowski, 2010). Además, se explica la minimización de la función suma del error cuadrático (SSE) y *Matlab* emplea como función a minimizar el error cuadrático medio (MSE), aunque en ambos casos el procedimiento algorítmico es análoga. En *Matlab* ya vienen implementados estos algoritmos, para activarlos y entrenar la red por épocas, simplemente se tiene que ejecutar los siguientes comandos:

a. Para ajuste de funciones:

```
net = fitnet(hiddenLayerSize);
[net, tr]=train(net, Inputs, Targets);
```

b. Para reconocimiento de patrones y clasificación:

```
net = patternnet(hiddenLayerSize, 'trainscg');  
[net, tr] = train(net, Inputs, Targets);
```

Donde *net* hace referencia a la red creada, *tr* es el conjunto de características de la red entrenada, *trainscg* es la función de entrenamiento, *Inputs* son las entradas y *Targets* las salidas objetivo.

2.5. Criterios de parada de entrenamiento

En *Matlab* se distinguen cinco criterios distintos para detener el proceso de entrenamiento, para activarlos y entrenar la red, simplemente se tiene que ejecutar los siguientes comandos:

- **Épocas**

El código que permite controlar el número de épocas (iteraciones del entrenamiento) máximo es:

```
net.trainParam.epochs = ... ;
```

- **Tiempo**

```
net.trainParam.time = ... ;
```

Permite introducir el máximo tiempo en segundos que se desea que dure el entrenamiento.

- **Rendimiento**

El código que permite fijar un valor límite para la función de optimización (MSE, en nuestro caso) es:

```
net.trainParam.goal = ... ;
```

Cuando, tras una iteración, se calcula el error entre la salida de la red y la salida objetivo, y su valor es menor o igual que el establecido mediante este parámetro, el entrenamiento se detiene, pues se considera que se ha alcanzado un valor óptimo.

- **Gradiente**

Para elegir el valor de gradiente a partir del cual se desea que el entrenamiento pare se utiliza:

```
net.trainParam.min_grad = ... ;
```


Durante el proceso de entrenamiento llega un punto en que el gradiente es tan pequeño que no tiene sentido continuar con el proceso ya que se consumen tiempo y recursos sin conseguir grandes disminuciones del error.

- **Comprobaciones de validación**

Se puede establecer el número de comprobaciones que se quiere que se hagan de la red antes de parar el entrenamiento mediante el conjunto de datos de validación. Esta herramienta permite controlar el sobreajuste de la red, que supone una pérdida de generalidad. Se aplica la técnica conocida como ‘*early stopping*’: Cuando se observa que, tras una serie de iteraciones, el error ha comenzado a crecer, se interrumpe el entrenamiento y se recuperan para los pesos los valores de la iteración que generaba menor error.

$$net.trainParam.max_fail = \dots ;$$

2.6. Procesos de simulación y/o pronóstico

La función ‘*sim*’ es parecida a la función ‘*train*’ pero que no actualizará los pesos de la red neuronal. Una vez que se tenga entrenada la red neuronal y esta ofrezca un resultado válido, se utiliza esta función para analizar nuevos patrones de entrada para ver si ha aprendido a responder correctamente. Es decir, se puede evaluar los pronósticos de la red, dado el conjunto de datos de *test* (*Input_Test*), así, comprobar con el conjunto de datos de *test* (*Targets_Test* - objetivo de *test*), finalmente graficar los errores resultantes:

$$y_p = sim(net, Inputs_Test);$$

ANEXO 3. Metodología de análisis *wavelet* mediante *Matlab*

3.1. Familia de *wavelets* y funciones de escala

El *software* “*Wavelet Toolbox*” incluye un gran número de *wavelets* que puede utilizar tanto para análisis continuos como discretos. Para el análisis discreto, los ejemplos incluyen *wavelets* ortogonales y *wavelets* biortogonales. Para el análisis continuo, el *software* “*Wavelet Toolbox*” incluye *Morlet*, *Meyer*, derivado de *Gaussian*, y *wavelets* de *Paul*, para mayor detalle revisar [Misiti et al. \(2016\)](#).

La elección de la *wavelet* está dictada por las características de la señal y la naturaleza de la aplicación. Si entiende las propiedades del análisis y síntesis *wavelet*, puede elegir una *wavelet* optimizada para su aplicación.

Las familias *wavelets* varían en términos de varias propiedades importantes. Ejemplos incluyen:

- Soporte de la *wavelet* en tiempo y frecuencia y tasa de decaimiento.
- Simetría o antisimetría de la *wavelet*. Los filtros de reconstrucción perfectos acompañantes tienen fase lineal.
- Número de momentos de desvanecimiento. *Wavelets* con un número creciente de momentos de desvanecimiento resultan en escasas representaciones para una gran clase de señales.
- Regularidad de la *wavelet*. Las *wavelets* más suaves proporcionan una resolución de frecuencia más nítida. Además, los algoritmos iterativos para la construcción de *wavelets* convergen más rápido.
- Existencia de una función de escala, φ .

Al ingresar *waveinfo* en la línea de comandos se muestra un reporte de las principales propiedades de familias *wavelet* disponibles. Para una familia de *wavelets* específica, utilice *waveinfo* con el nombre abreviado de la familia *wavelet*. Puede encontrar los nombres abreviados de la familia *wavelet* que se enumeran en la siguiente tabla y en la página de referencia de *waveinfo*.

Nombre corto de familia wavelet	Nombre de familia wavelet
'haar'	<i>Haar wavelet</i>
'db'	<i>Daubechies wavelets</i>
'sym'	<i>Symlets</i>
'coif'	<i>Coiflets</i>
'bior'	<i>Biorthogonal wavelets</i>
'rbio'	<i>Reverse biorthogonal wavelets</i>
'meyr'	<i>Meyer wavelet</i>
'dmey'	<i>Discrete approximation of Meyer wavelet</i>
'gaus'	<i>Gaussian wavelets</i>
'mexh'	<i>Mexican hat wavelet</i>
'morl'	<i>Morlet wavelet</i>
'cgau'	<i>Complex Gaussian wavelets</i>
'shan'	<i>Shannon wavelets</i>
'fbsp'	<i>Frequency B-Spline wavelets</i>
'cmor'	<i>Complex Morlet wavelets</i>
'fk'	<i>Fejer-Korovkin wavelets</i>

Para mostrar información detallada sobre las *wavelets* ortogonales menos asimétricas de *Daubechies*, ingrese:

```
waveinfo('sym')
```

Para calcular la función *wavelet* y escala (si está disponible), use *wavefun*. No hay función de escala asociada con la *wavelet Morlet*. Para calcular la *wavelet Morlet*, puede ingresar:

```
[psi,xval] = wavefun('morl',10);
plot(xval,psi); title('Morlet Wavelet');
```

Para las *wavelets* asociadas con un análisis multiresolución, puede calcular tanto la función de escala y la *wavelet*. El siguiente código devuelve la función de escala y *wavelet* para la *wavelet* de la fase extrema de *Daubechies* con 4 momentos de desvanecimiento.

```
[phi, psi, xval] = wavefun('db4',10);
subplot(211);
plot(xval, phi);
title('db4 Scaling Function');
subplot(212);
plot(xval, psi);
title('db4 Wavelet');
```

3.2. Análisis *wavelet* discreto unidimensional

Esta sección describe las características del análisis de *wavelet* discreto unidimensional usando la “*Wavelet Toolbox*” de *Matlab*. La “*Wavelet Toolbox*” proporciona estas funciones para el análisis de la señal unidimensional.

Funciones de análisis de descomposición

Nombre de la función	Propósito
dwt	Descomposición de un solo nivel
wavedec	Descomposición
wmaxlev	Máximo nivel de descomposición wavelet

Funciones de síntesis de reconstrucción

Nombre de la función	Propósito
idwt	Reconstrucción de un solo nivel
waverec	Reconstrucción completa
wrcoef	Reconstrucción selectiva
upcoef	Reconstrucción única

Utilidades de la estructura de descomposición

Nombre de la función	Propósito
detcoef	Extracción de coeficientes de detalle
appcoef	Extracción de coeficientes de aproximación
upwlev	Recomposición de la estructura de descomposición

Eliminación de ruido y compresión

Nombre de la función	Propósito
ddencomp	Proporciona valores por defecto para la elim. de ruidos y compresión
wbmpen	Umbral penalizado para la eliminación de ruido wavelet 1-D o 2-D
wdcbm	Umbrales para wavelet 1-D usando la estrategia de Birgé-Massart
wdencomp	Eliminación de ruido y compresión wavelet
wden	Eliminación de ruido wavelet automática
wthrmngr	Administrador de configuración de umbral

a. Análisis unidimensional utilizando la línea de comandos

Este ejemplo implica una señal del mundo real - consumo eléctrico medido en el transcurso de 3 días. Esta señal es particularmente interesante debido al ruido introducido cuando se desarrolló un defecto en el equipo de monitorización a medida que se realizaban las mediciones. El análisis *wavelet* elimina de forma eficaz el ruido.

- a.1.** Cargue la señal y seleccione una porción para el análisis *wavelet*.

```
load leleccum;  
s = leleccum(1:3920);  
l_s = length(s);
```

- a.2.** Realizar una descomposición *wavelet* de un solo nivel de una señal.

Realice una descomposición de un solo nivel de la señal utilizando *wavelet db1*.
Escriba:

```
[cA1,cD1] = dwt(s,'db1');
```

Esto genera los coeficientes del nivel 1 de aproximación (cA1) y detalle (cD1).

- a.3.** Construya aproximaciones y detalles a partir de los coeficientes.

Para construir la aproximación y el detalle del nivel 1 (A1 y D1) a partir de los coeficientes cA1 y cD1, escriba

```
A1 = upcoef('a',cA1,'db1',1,l_s);  
D1 = upcoef('d',cD1,'db1',1,l_s);  
o  
A1 = idwt(cA1,[],'db1',l_s);  
D1 = idwt([],cD1,'db1',l_s);
```

- a.4.** Muestra la aproximación y el detalle.

Para mostrar los resultados de la descomposición de nivel uno, escriba

```
subplot(1,2,1); plot(A1); title('Approximation A1')  
subplot(1,2,2); plot(D1); title('Detail D1')
```

- a.5.** Regenere una señal utilizando la transformada *wavelet* inversa.

Para encontrar la transformada inversa, ingrese

```
A0 = idwt(cA1,cD1,'db1',ls);  
err = max(abs(s-A0))
```

- a.6.** Realizar una descomposición *wavelet* multinivel de una señal.

Para realizar una descomposición de nivel 3 de la señal (otra vez usando la *wavelet db1*), escriba

```
[C,L] = wavedec(s,3,'db1');
```

Los coeficientes de todos los componentes de una descomposición de tercer nivel (es decir, la aproximación de tercer nivel y los primeros tres niveles de detalle) se devuelven concatenados en un vector, C. El vector L da las longitudes de cada componente.

- a.7.** Extraer los coeficientes de aproximación y de detalle.

Para extraer los coeficientes de aproximación de nivel 3 de C, escriba

```
cA3 = appcoef(C,L,'db1',3);
```

Para extraer los coeficientes de detalle de los niveles 3, 2, y 1 de C, escriba

```
cD3 = detcoef(C,L,3);
```

```
cD2 = detcoef(C,L,2);
```

```
cD1 = detcoef(C,L,1);
```

o

```
[cD1,cD2,cD3] = detcoef(C,L,[1,2,3]);
```

- a.8.** Reconstruya la aproximación de nivel 3 y los detalles de nivel 1, 2 y 3.

Para reconstruir la aproximación de nivel 3 a partir de C, escriba

```
A3 = wrcoef('a',C,L,'db1',3);
```

Para reconstruir los detalles en los niveles 1, 2 y 3, desde C, escriba

```
D1 = wrcoef('d',C,L,'db1',1);
```

```
D2 = wrcoef('d',C,L,'db1',2);
```

```
D3 = wrcoef('d',C,L,'db1',3);
```

- a.9.** Muestra los resultados de una descomposición multinivel.

Para mostrar los resultados de la descomposición de nivel 3, escriba

```
subplot(2,2,1); plot(A3);
```

```
title('Aproximación A3')
```

```
subplot(2,2,2); plot(D1);
```

```
title('Detalle D1')
```

```
subplot(2,2,3); plot(D2);
```

```
title('Detalle D2')
```

```
subplot(2,2,4); plot(D3);
```

```
title('Detalle D3')
```

a.10. Reconstruya la señal original a partir de la descomposición de nivel 3.

Para reconstruir la señal original de la estructura de descomposición *wavelet*, escriba

```
A0 = waverec(C,L,'db1');  
err = max(abs(s-A0))
```

a.11. Eliminación de ruido crudo de una señal.

Usar *wavelets* para eliminar el ruido de una señal requiere identificar qué componente o componentes contienen el ruido y luego reconstruir la señal sin esos componentes.

En este ejemplo, se observa que las aproximaciones sucesivas se hacen cada vez menos ruidosas a medida que más y más información de alta frecuencia es filtrada fuera de la señal.

La aproximación de nivel 3, A3, es bastante limpia como una comparación entre ello y la señal original.

Para comparar la aproximación a la señal original, escriba

```
subplot(2,1,1);plot(s);title('Original'); axis off  
subplot(2,1,2);plot(A3);title('Nivel 3 Aproximación');  
axis off
```

Por supuesto, al descartar toda la información de alta frecuencia, también se ha perdido muchas de las características más nítidas de la señal original.

La eliminación de ruido óptima requiere un enfoque más sutil llamado umbralización (*thresholding*). Esto implica descartar sólo la parte de los detalles que excede un cierto límite.

a.12. Eliminar el ruido mediante el umbral.

Se ve nuevamente los detalles de nuestro análisis de nivel 3.

Para mostrar los detalles D1, D2 y D3, escriba

```
subplot(3,1,1); plot(D1); title('Detalle Nivel 1'); axis off  
subplot(3,1,2); plot(D2); title('Detalle Nivel 2'); axis off  
subplot(3,1,3); plot(D3); title('Detalle Nivel 3'); axis off
```

La mayor parte del ruido se produce en la última parte de la señal, donde los detalles muestran su mayor actividad. ¿Qué pasa si se limita la fuerza de los detalles al

restringir sus valores máximos? Esto tendría el efecto de reducir el ruido sin afectar los detalles durante la mayor parte de su duración. Pero hay una mejor manera.

Tenga en cuenta que *cD1*, *cD2* y *cD3* son sólo vectores *Matlab*, por lo que se podría manipular directamente cada vector, configurando cada elemento en alguna fracción del valor máximo o promedio de los vectores. Entonces se podría reconstruir nuevas señales de detalle *D1*, *D2* y *D3* a partir de los coeficientes umbralizados.

Para eliminar el ruido de la señal, utilice el comando *ddencmp* para calcular los parámetros por defecto y el comando *wdencmp* para llevar a cabo la eliminación de ruido real, escriba

```
[thr,sorh,keepapp] = ddencmp('den','wv',s);  
clean = wdencmp('gbl',C,L,'db1',3,thr,sorh,keepapp);
```

Tenga en cuenta que *wdencmp* utiliza los resultados de la descomposición (*C* y *L*) que se calcula en el paso a.6. También se especifica que se utiliza la *wavelet* *db1* para realizar el análisis original y se especifica la opción de umbral global '*gbl*'. Consulte *ddencmp* y *wdencmp* en las páginas de referencia para obtener más información sobre el uso de estos comandos.

Para visualizar tanto la señal original como la señal sin ruido, escriba

```
subplot(2,1,1); plot(s(2000:3920)); title('Original')  
subplot(2,1,2); plot(clean(2000:3920)); title('Sin ruido')
```

Aquí sólo se ha trazado la última parte ruidosa de la señal. Observe cómo se ha eliminado el ruido sin comprometer el detalle nítido de la señal original. Esta es una fortaleza del análisis *wavelet*.

Si bien el uso de las funciones de línea de comandos para eliminar el ruido de una señal puede ser engorroso, las herramientas de interfaz gráfica del *software* incluyen una función de eliminación de errores fácil de usar que incluye el umbral automático.

ANEXO 4. Código *Matlab* del modelo de redes neuronales artificiales

```
=====
% MODELOS DE REDES NEURONALES ARTIFICIALES (p.ej. RNA4)
% Escuela de Posgrado de la UNALM
% Maestría en Ingeniería de Recursos Hídricos
% Autor: Ing. Lucio E. Vergara Saturno
% E-mail: versat.irh@gmail.com
% Fecha: 15 de noviembre del 2018
=====

% Remoción de figuras, datos y borrado de la pantalla
    close all;
    clear all;
    clc;

=====
% Cargado de datos y preprocesamientos
=====

% Cargado de datos de caudales diarios (p.ej. Est. Tamshiyacu)
    load 'D:\tesis_maestria_versat\tamsh.txt';
    qdaily = tamsh;

% Definición del periodo de trabajo de los datos
    qd_all = qdaily(366:10227,1);

% Transformación de los datos en [-1 a 1] <-- (p.ej. tansig)
    [prel,PS] = mapminmax(qd_all.',-1,1);
    qd_tt = mapminmax('apply',qd_all,PS);

% Partición de los datos en subseries de train y tests
    qd_train = qd_tt(1:7670,1);
    qd_tests = qd_tt(7671:9862,1);

% Número total de entradas y horizonte de pronóstico
    cln = 25;
    cln2= cln-1;

% Control de generación de números aleatorios
    rng(0, 'v5uniform')

=====
% Fase de entrenamiento (train)
=====

% Creación de la matriz con los rezagos (lags)
    mat1 = lagmatrix(qd_train,0:cln2);

% Reconfigura la matriz de entrada y borra datos NaN
    matriz_train = fliplr(mat1(cln:end,:));

% Entradas(x) y salida(t) para el entrenamiento del RNA
    x = matriz_train(:,1:4);
    t = matriz_train(:,cln);
```

```

% Matriz transpuesta para el formato de entrada del RNA
T = x';
P = t';

% Creación de la estructura del modelo RNA
net = feedforwardnet(8, 'trainlm');

% Entrenamiento del modelo RNA
net = train(net,T,P);

% Evaluación del modelo RNA con datos de entrenamiento
train_val = sim(net,T);

% Matriz transpuesta de los resultados, entrenamiento
y_train = train_val';

% Revertiendo la transformación
t = mapminmax('reverse',t,PS);
y_train = mapminmax('reverse',y_train,PS);

% Cálculo de la bondad de ajuste del modelo RNA, train
gf_train = gfit2(t(:,1),y_train(:,1),'all','v');

%=====
% Fase de validación (tests)
%=====

% Creación de la matriz con los rezagos (lags)
mat2 = lagmatrix(qd_tests,0:cln2);

% Reconfigura la matriz de entrada y borra datos NaN
matriz_test = fliplr(mat2(cln:end,:));

% Entradas(xt) y salida(tt) para el testeo del RNA
xt = matriz_test(:,1:4);
tt = matriz_test(:,cln);

% Matriz transpuesta para el formato de entrada del RNA
testT = xt';
testP = tt';

% Evaluación del modelo RNA con datos de testeo
test_val = sim(net,testT);

% Matriz transpuesta de los resultados, tests
y_test = test_val';

% Revertiendo la transformación
tt = mapminmax('reverse',tt,PS);
y_test = mapminmax('reverse',y_test,PS);

% Cálculo de la bondad de ajuste del modelo RNA, tests
gf_test = gfit2(tt(:,1),y_test(:,1),'all','v');

```

ANEXO 5: Código *Matlab* del modelo híbrido *wavelet* redes neuronales

```
%=====
% MODELOS HÍBRIDOS WAVELET REDES NEURONALES (p.ej. WRN12)
% Escuela de Posgrado de la UNALM
% Maestría en Ingeniería de Recursos Hídricos
% Autor: Ing. Lucio E. Vergara Saturno
% E-mail: versat.irh@gmail.com
% Fecha: 15 de noviembre del 2018
%=====

% Remoción de figuras, datos y borrado de la pantalla
    close all;
    clear all;
    clc;

%=====
% Cargado de datos y preprocesamientos
%=====

% Cargado de datos de caudales diarios (p.ej. Est. Tamshiyacu)
    load 'D:\tesis_maestria_versat\tamsh.txt';
    qdaily = tamsh;

% Definición del periodo de trabajo de los datos
    qd_all = qdaily(366:10227,1);

% Transformación de los datos en [-1 a 1] <-- (p.ej. tansig)
    [prel,PS] = mapminmax(qd_all,-1,1);
    qd_tt = mapminmax('apply',qd_all,PS);

% Partición de los datos en subseries de train y tests
    qd_train = qd_tt(1:7670,1);
    qd_tests = qd_tt(7671:9862,1);

% Número total de entradas y horizonte de pronóstico
    cln = 34;
    cln2= cln-1;

% Control de generación de números aleatorios
    rng(0,'v5uniform')

%=====
% Fase de entrenamiento (train)
%=====

% Creación de la matriz con los rezagos (lags)
    mat1 = lagmatrix(qd_train,0:cln2);

% Reconfigura la matriz de entrada y borra datos NaN
    matriz_train = fliplr(mat1(cln:end,:));

% Entradas(x) y salida(t) para el entrenamiento del WRN
    x = matriz_train(:,1:4);
    t = matriz_train(:,cln);
```

```

% Aplicación de la transformada wavelet discreta (TWD), train
nd = 4;
for k = 1:nd
    [C,L] = wavedec(x(:,k),nd,'db2');
    A = wrcoef('a',C,L,'db2',nd);
    for s = 1:nd
        Dp(:,s) = wrcoef('d',C,L,'db2',s);
    end
    D = fliplr(Dp(:,:));
    var = ['wm',int2str(k),'= cat(2,A,D)'];
    eval(var)
    clear A C Dp D L
end
wme = cat(2,wm1,wm2,wm3,wm4);
x = wme;

% Matriz transpuesta para el formato de entrada del WRN, train
T = x';
P = t';

% Creación de la estructura del modelo híbrido WRN
net = feedforwardnet(7,'trainlm');

% Entrenamiento del modelo híbrido WRN
net = train(net,T,P);

% Evaluación del modelo WRN con datos de entrenamiento
train_val = sim(net,T);

% Matriz transpuesta de los resultados, entrenamiento
y_train = train_val';

% Revertiendo la transformación
t = mapminmax('reverse',t,PS);
y_train = mapminmax('reverse',y_train,PS);

% Cálculo de la bondad de ajuste del modelo híbrido WRN, train
gf_train = gfit2(t(:,1),y_train(:,1),'all','v');

%=====
% Fase de validación (tests)
%=====

% Creación de la matriz con los rezagos (lags)
mat2 = lagmatrix(qd_tests,0:cln2);

% Reconfigura la matriz de entrada y borra datos NaN
matriz_test = fliplr(mat2(cln:end,:));

% Entradas(xt) y salida(tt) para el testeo del WRN
xt = matriz_test(:,1:4);
tt = matriz_test(:,cln);

% Aplicación de la transformada wavelet discreta (TWD), tests
nd = 4;

```

```

for k = 1:nd
    [C,L] = wavedec(xt(:,k),nd,'db2');
    A = wrcoef('a',C,L,'db2',nd);
    for s = 1:nd
        Dp(:,s) = wrcoef('d',C,L,'db2',s);
    end
    D = fliplr(Dp(:,:));
    var = ['wm',int2str(k),' = cat(2,A,D)'];
    eval(var)
    clear A C Dp D L
end
wmt = cat(2,wm1,wm2,wm3,wm4);
xt = wmt;

% Matriz transpuesta para el formato de entrada del WRN, tests
testT = xt';
testP = tt';

% Evaluación del modelo WRN con datos de testeo
test_val = sim(net,testT);

% Matriz transpuesta de los resultados, tests
y_test = test_val';

% Revertiendo la transformación
tt = mapminmax('reverse',tt,PS);
y_test = mapminmax('reverse',y_test,PS);

% Cálculo de la bondad de ajuste del modelo híbrido WRN, tests
gf_test = gfit2(tt(:,1),y_test(:,1),'all','v');

```