



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

máster en ingeniería
hidráulica y medio ambiente
mihma

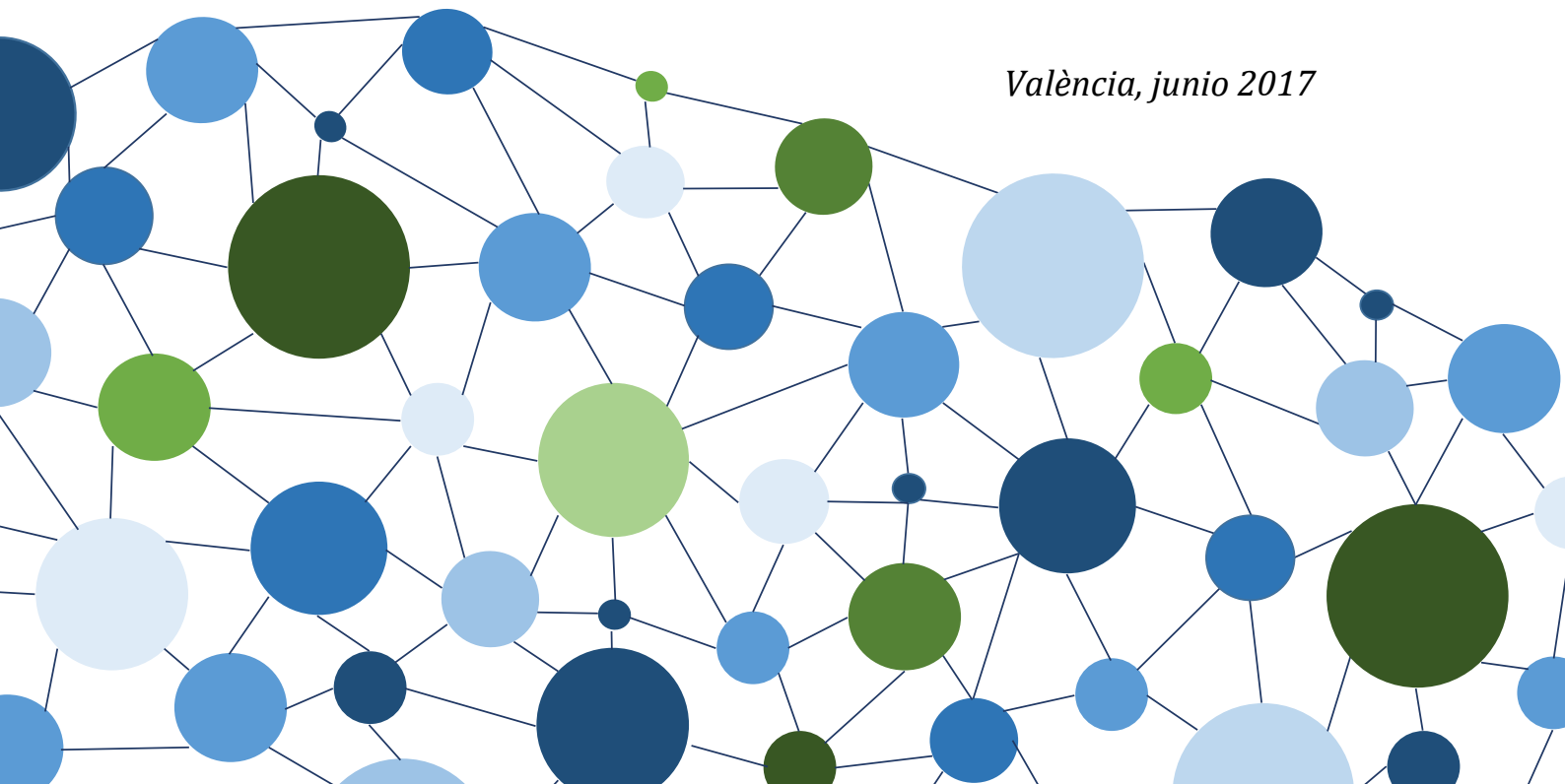
Trabajo Fin de Máster

USO DE REDES NEURONALES ARTIFICIALES PARA LA MODELACIÓN DEL ESTADO TRÓFICO DE LAS MASAS DE AGUA, APLICADO AL PARQUE NATURAL DE LA ALBUFERA DE VALÈNCIA

Máster en Ingeniería Hidráulica y Medio Ambiente
**Intensificación: ORDENACIÓN, RESTAURACIÓN Y GESTIÓN DE
CUENCAS**

Autor: LAURA RAMOS SOLER
Tutor: Dr. Javier Paredes Archiola
Cotutor/es: Dr. Joaquín Andreu Álvarez,
Edgar Belda Ibáñez

València, junio 2017





Resumen del Trabajo de Fin de Máster

Datos del proyecto
Título del TFM en español: Uso de Redes Neuronales Artificiales para la modelación del estado trófico de las masas de aguas, aplicado al Parque Natural de la Albufera de València
Título del TFM en inglés: Use of Artificial Neural Networks to modeling the trophic state of a water body, applied to the Albufera Natural Park of València
Título del TFM en Valenciano: Utilització de Xarxes Neuronals Artificials per a modelar l'estat tròfic d'una massa d'aigua, aplicada al Parc Natural de l'Albufera de València
Alumno: Laura Ramos Soler
Tutor: Dr. Javier Paredes Arquiloa
Cotutor/es: Dr. Joaquín Andreu Álvarez Edgar Belda Ibañez
Fecha de Lectura: junio, 2017

Resumen

En español (máximo 5000 caracteres)

La alteración del estado trófico de las masas de agua es uno de los principales problemas que afecta a las zonas húmedas del planeta. Entre los diversos procesos que modifican la calidad del agua, cabe destacar la eutrofización, que consiste en el crecimiento anormal de las poblaciones de fitoplancton provocado por vertidos con altas concentraciones de nutrientes.

Actualmente, el control del estado trófico de las aguas supone un reto para los gestores de las cuencas hidrográficas y en sus manos está el intentar minimizar y evitar el efecto de las actividades antrópicas sobre los ecosistemas acuáticos. Para mejorar la toma de decisiones, los actores responsables utilizan herramientas de cálculo que facilitan el conocimiento del sistema y la respuesta del mismo ante diversos escenarios. Entre dichas herramientas destacan, por un lado, los modelos clásicos mecanicistas basados en la conservación de la masa. Por otro lado, los modelos conexionistas basados en unidades sencillas interconectadas con aprendizaje en base a múltiples patrones.

El principal objetivo de este Trabajo Final de Máster ha sido analizar el uso de las Redes Neuronales Artificiales para la modelización de las concentraciones de clorofila-*a*. Las RNA se aplican en el ámbito de los recursos hídricos desde los años noventa, resolviendo problemas tanto de cantidad como de calidad de las aguas. Los resultados satisfactorios de los últimos trabajos

publicados, pone de manifiesto la utilidad de estas estructuras para formar sistemas simples capaces de modelar problemas complejos.

A fin de comprobar el funcionamiento de estos modelos, se han aplicado dichas técnicas en el ámbito del Parque Natural de la Albufera, paraje caracterizado por un elevado grado de eutrofización. Para alcanzar el objetivo propuesto, se ha realizado un análisis de sensibilidad para identificar cuáles son las variables que afectan directamente a las concentraciones de clorofila-a de la laguna. También se ha trabajado con diferentes tipologías estructurales de RNA para poder determinar cómo afectan estos aspectos técnicos a la estimación de las concentraciones. Finalmente se han recopilado los resultados y se han extraído unas conclusiones.

En valenciano (máximo 5000 caracteres)

L'alteració de l'estat tròfic de les masses d'aigua és un dels principals problemes que afecten a les zones humides del planeta. Entre els diversos processos que modifiquen la qualitat de l'aigua, cal destacar l'eutrofització, que consisteix en el creixement anormal de les poblacions de fitoplàncton provocat per abocaments amb altes concentracions de nutrients.

Actualment, el control de l'estat tròfic de les aigües suposa un repte per als gestors de les conques hidrogràfiques i a les seues mans hi ha el intentar minimitzar i evitar l'efecte de les activitats antròpiques en els ecosistemes aquàtics. Per-a millorar la presa de decisions, els actors responsables utilitzen eines de càlcul que faciliten el coneixement del sistema i la resposta del mateix davant diversos escenaris. Entre aquestes eines destaquen, els models clàssics mecanicistes basats en la conservació de la massa i els models connexionistes basats en unitats senzilles interconnectades amb aprenentatge basant-se múltiples patrons.

El principal objectiu d'aquest Treball Final de Màster ha sigut analitzar l'ús de les Xarxes Neuronals Artificials per a la modelització de les concentracions de clorofil·la-a. Les xarxes neuronals artificials s'apliquen en l'àmbit dels recursos hídrics des dels anys noranta, resolent problemes tant de quantitat com de qualitat de les aigües. Els resultats satisfactoris dels últims treballs publicats, posa de manifest la utilitat d'aquestes estructures per formar sistemes simples capaços de modelar problemes complexos.

Per tal de comprovar el funcionament d'aquests models, s'ha aplicat aquestes tècniques en l'àmbit del Parc Natural de l'Albufera, paratge caracteritzat per un elevat grau d'eutrofització. Per arribar a l'objectiu proposat, s'ha realitzat un anàlisi de sensibilitat per identificar quines són les variables que afecten directament a les concentracions de clorofil·la-a de la llacuna. També s'analitzaran les diferents tipologies estructurals de RNA per poder determinar com afecten aquests aspectes tècnics a l'estimació de les concentracions. Finalment se han recopilat els resultats i s'ha arribat a unes conclusions.

En inglés (máximo 5000 caracteres)

The alteration of the trophic state of the water bodies is one of the main problems affecting the wetlands of the planet. Among the various processes that modify water quality, highlight eutrophication, which consists in the atypical growth of phytoplankton populations caused by discharges with high concentrations of nutrients.

At present, the control of the trophic state of the waters is a challenge for the managers of the watersheds and is their responsibility attempt to minimize and avoid the effect of the anthropic activities on the aquatic ecosystems. To improve decision-making, responsible actors use calculation tools that facilitate the knowledge of how works the systems and its response to different scenarios. Among these tools stand out, the classic mechanical models based on the conservation of mass and the connectionist models based on simple units interconnected with learning based on multiple patterns.

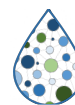
The main objective of this Final Project is to analyse the use of Artificial Neural Networks for the modelling of chlorophyll-a concentrations. ANNs have been applied in the field of water resources since the 1990s, solving problems of quantity and quality of water. The satisfactory results of the last published works show the usefulness of these structures to form simple systems capable of modelling complex problems.

In order to verify the operation of these models, this technique will be applied in the field of the Albufera Natural Park, a place characterized by a high level of eutrophication. To reach the proposed objective, a sensitivity analysis will be implement to identify the variables that directly affect the concentrations of chlorophyll-a in the water body. We will also analyse the different structural typologies of RNA in order to determine how these technical aspects affect the estimation of concentrations.

Palabras clave español (máximo 5): Calidad de aguas, modelación, Redes Neuronales Artificiales, eutrofización

Palabras clave valenciano (máximo 5): Qualitat de l'aigua, modelació, Xarxes Neuronals Artificials, eutrofització

Palabras clave inglés (máximo 5): Water quality, modeling, Artificial Neural Networks, eutrophication



Índice de la Memoria

1.	Introducción.....	1
1.1	Motivación del trabajo.....	2
2.	Objetivos.....	5
3.	Estado del Arte.....	7
3.1	Introducción.....	7
3.1.1	<i>Definición y descripción de RNAs:</i>	7
3.1.2	<i>Implementación de una red neuronal:</i>	10
3.2	Ventajas e Inconvenientes de las RNAs.....	11
3.3	RNA y Recursos Hídricos.....	16
3.4	RNA y Calidad de las aguas.....	17
4.	Metodología.....	25
4.1	Software.....	26
4.2	Tipo de redes.....	27
4.3	Pretratamiento de los datos.....	29
4.4	Algoritmos de entrenamiento.....	31
4.5	Funciones de activación.....	32
4.5.1	<i>Funciones sigmoidales:</i>	34
4.6	Estructura de la red.....	35
4.7	Validación de la RNA.....	36
4.8	Análisis de Resultados:.....	37
5.	Caso de Estudio.....	39
5.1	Descripción Parque Natural de la Albufera.....	39
5.1.1	<i>Localización</i>	41
5.1.2	<i>Eutrofización de la laguna</i>	42
5.2	Modelación clorofila- <i>a</i>	45
5.3	Escenarios propuestos.....	47
5.4	Muestra de datos.....	48
5.5	Resultados.....	53
5.5.1	<i>Escenario_01</i>	53
5.5.2	<i>Escenario_02</i>	58
5.5.3	<i>Escenario_03</i>	62

5.5.4	<i>Escenario_04</i>	67
5.5.5	<i>Comparación escenarios</i>	74
6.	Conclusiones	77
7.	Futuras líneas de investigación.....	81
8.	Referencias	83
	ANEJOS.....	89
A.	CÓDIGO REDES MULTILAYER PERCEPTRON.....	89
B.	CÓDIGO RED RESUCRRENTE JORDAN.....	98



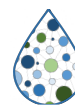
Índice de Gráficas

Gráfica 1 Error Cuadrático Medio de una RNA variando el número de neuronas oculta. Fuente: Elaboración propia.....	15
Gráfica 2 funciones sigmoidales: (a) logística; (b) tangente hiperbólica.....	34
Gráfica 3 Relación entre la evolución del error del entrenamiento y el error de la validación (Marín Diazaraque, 2007)	37
Gráfica 4 Resultados de la calibración de Humecal en la Albufera- 2017.....	48
Gráfica 5 Coeficiente de Pearson de los posibles inputs.....	49
Gráfica 6 Rango del MSE de diez redes (variando el número de neuronas ocultas)	54
Gráfica 7 Ajuste de la RNA-Escenario_01	56
Gráfica 8 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_01	57
Gráfica 9 Representación datos observados, resultados Humecal y resultados RNA_01.....	57
Gráfica 10 Red Neuronal Artificial- Escenario_02.....	59
Gráfica 11 Ajuste de la RNA-Escenario_02.....	60
Gráfica 12 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_02.....	61
Gráfica 13 Representación datos observados, resultados Humecal y resultados RNA_02.....	62
Gráfica 14 Ajuste de la RNA-Escenario_03.....	64
Gráfica 15 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_03.....	65
Gráfica 16 Representación datos observados, resultados Humecal y resultados RNA_03 de la muestra de validación	66
Gráfica 17 Red Neuronal Artificial- Escenario_04.....	67
Gráfica 18 Ajuste de la RNA-Escenario_04.....	68
Gráfica 19 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_04.....	69
Gráfica 20 Representación datos observados, resultados Humcal y resultados RNA_04 de la muestra de validación	70

Gráfica 21 Ajuste de la RNA-Escenario_04.....	72
Gráfica 22 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_05.....	73
Gráfica 23 Representación datos observados, resultados Humecal y resultados RR_04.....	74
Gráfica 24 RMSE y r^2 de las cinco RNA.....	75
Gráfica 25 Resultados de los cuatro escenarios y datos observados de clorofila-a.....	76

Índice de Ilustraciones

Ilustración 1 Estructura básica de una RNA (Bravo-Oviedo & Kindermann, 2004)..	9
Ilustración 2 Esquema de la metodología para la construcción de una RNA : (Gestal, 2009)	25
Ilustración 3 Topología de red neuronal con conexiones hacia delante (Bravo-Oviedo & Kindermann, 2004)	29
Ilustración 4 Esquema del funcionamiento del código para la elección de la estructura neuronal. Fuente: Elaboración propia.....	36
Ilustración 5 Imagen satélite de la Albufera de Valencia. Fuente: Image Landsat/Google Earth.....	41
Ilustración 6 Principales factores y procesos que desencadenan y mantienen las respuestas del proceso de eutrofización (Mazzeo et al., 2002)	43
Ilustración 7 Esquema del modelo Humecal (Belda, 2015)	46
Ilustración 8 Ciclo del nitrógeno en el agua.(Las líneas discontinuas indican los procesos que se desarrollan en condiciones anaeróbicas), (Chapra, 1997).....	51
Ilustración 9 Red Neuronal Artificial- Escenario_01.....	55
Ilustración 10 Red Neuronal Artificial- Escenario_03	63
Ilustración 11 Red Neurona Artificial Jordan- Escenario_04.....	71



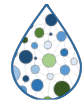
Índice de Tablas

Tabla 1 Escenarios propuestos (Cho, 2014)	20
Tabla 2 r^2 , RMSE, NSE y % de Diferencia entre las predicciones y los datos observados de clorofila-a (Cho, 2014)	21
Tabla 3 Resultados estimación de clorofila-a dependiendo los inputs.....	23
Tabla 4 Caracterización del estado trófico de un ecosistema lacustre	44
Tabla 5 Clasificación del estado trófico del ecosistema en función del TSI (Carlson, 1977)	45
Tabla 6 Leyenda grafica 5.....	49
Tabla 7 Pesos de la RNA-Escenario_01	55
Tabla 8 Pesos de la RNA-Escenario_02	59
Tabla 9 Pesos de la RNA-Escenario_03	64
Tabla 10 Pesos de la RNA-Escenario_04.....	68

Abreviaturas

ARMA	Modelo Autorregresivo Integrado de Promedio Móvil
CE	Conductividad Efectiva
DBO	Demanda Biológica de Oxígeno
DO	Demanda de Oxígeno
DQO	Demanda Química de Oxígeno
EDAR	Estación Depuradora de aguas Residuales
EPA	The US Environmental Protection Agency
MLP	<i>Multilayer Perceptron</i>
MSE	Error Cuadrático Medio
OD	Oxígeno Disuelto
PC	Ficocianina
PNAV	Parque Natural de la Albufera de Valencia
r	Coefficiente de Correlación de Pearson
r^2	Coefficiente de determinación
RBF-NN	Radial Basis Function- Neural Networks
RLM	Regresión Lineal Multivariante

RMSE	Raíz cuadrada del Error Cuadrático Medio
RNA	Redes Neuronales Artificiales
RNN	Redes Neuronales Recurrentes (<i>Recurrent Neural Networks</i>)
RRHH	Recursos Hídricos
RSR	Ratio de Desviación Estándar
SSRRHH	Sistemas de Recursos Hídricos
SST	Sólidos Suspendedos Totales
ST	Sólidos Totales
SD	Sólidos Disueltos
T	Temperatura
TN	Nitrógeno Total
TP	Fósforo Total
TFM	Trabajo Final de Máster
WQI	Índice de Calidad de Agua (<i>Water Quality Index</i>)
ZEPA	Zona de Especial Protección para Aves



1. Introducción

Durante los últimos años, se han desarrollado numerosos modelos de calidad de agua para simular una amplia variedad de contaminantes en diversos tipos de masas de agua. En la naturaleza el agua no se encuentra pura, sino que incorpora diversas sustancias desde su caída a la superficie de la tierra hasta que llega al mar. La calidad del recurso afecta directamente a sus posibles usos. La reciente modelación conjunta de la cantidad y calidad del agua es una gestión integrada más acorde con la realidad (Paredes & Lund, 2006).

La alteración del estado trófico de las masas de agua es uno de los principales problemas que afecta a las zonas húmedas del planeta. Entre los diversos procesos que modifican la calidad del agua, cabe destacar la eutrofización, que consiste en el crecimiento anormal de las poblaciones de fitoplancton provocado por vertidos con altas concentraciones de nutrientes.

Durante los últimos años se ha ido reconociendo el valor ambiental de los humedales y se ha dedicado mucho esfuerzo en trasladar a los organismos ambientales competentes la necesidad de aplicar gestiones adecuadas que fomenten la conservación y el uso racional de estos espacios. El documento más relevante en este campo es el convenio RAMSAR, redactado durante la Convención Relativa a los Humedales de Importancia Internacional especialmente como Hábitat de Aves Acuáticas. Uno de los primeros acuerdos intergubernamentales que reconoce a los humedales como parajes de gran valor ambiental.

Por su gran riqueza en especies y por constar de una elevada variedad de hábitats, algunos humedales también están reconocidos por la Directiva 92/43/CEE del Consejo de 21 de mayo de 1992 relativa a la conservación de los hábitats naturales y de la fauna y flora silvestres, también conocida como Directiva Hábitats.

Actualmente, el control del estado trófico de las aguas supone un reto para los gestores de las cuencas hidrográficas y en sus manos está el intentar minimizar y evitar el efecto de las actividades antrópicas sobre los ecosistemas acuáticos. Para mejorar la toma de decisiones, los actores responsables utilizan herramientas de cálculo que facilitan el conocimiento del sistema y la respuesta del mismo ante diversos escenarios. Entre dichas herramientas destacan, por un lado, los modelos



clásicos mecanicistas basados en la conservación de la masa. Por otro lado, los modelos conexionistas basados en unidades sencillas interconectadas con aprendizaje en base a múltiples patrones.

Los buenos resultados obtenidos por los modelos conexionistas ponen el foco en el uso de estas técnicas para la modelación de problemas en los que comúnmente se han empleado modelos más clásicos. Por ello, en el presente trabajo se ha descrito y analizado los resultados obtenidos tras la implementación de Redes Neuronales Artificiales [RNA] en la simulación de las concentraciones de clorofila- α en la laguna del Parque Natural de la Albufera de Valencia [PNAV].

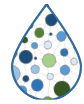
1.1 Motivación del trabajo

Según el Convenio RAMSAR, el Parque Natural de la Albufera consta de 4 puntos de los 8 existentes para ser catalogado dentro de la lista RAMSAR:

- **Criterio 2**, sustenta especies vulnerables, en peligro, en peligro crítico o comunidades ecológicas amenazadas, según la Directiva Hábitats.
- **Criterio 3**, sustenta poblaciones de especies vegetales y/o animales importantes para mantener la diversidad biológica en una región geográfica determinada.
- **Criterio 5**, sustenta de manera regular una población de 20000 o más aves acuáticas.
- **Criterio 6**, sustenta de manera regular 1% de los individuos de una población de una especie o subespecie de ave acuática.

Dentro de la lista RAMSAR se encuentran los humedales de mayor importancia internacional. Según el artículo 3.1 del Convenio, se establece que “las Partes Contratantes deberán elaborar y aplicar su planificación de forma que favorezca la conservación de los humedales incluidos en la Lista”.

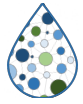
La eutrofización es uno de los problemas más graves que sufre el Parque Natural. Mediante el presente TFM se pretende contribuir a ampliar el conocimiento en la estimación de las concentraciones de clorofila- α , para facilitar de este modo la gestión por parte de los actores responsables y favorecer a la conservación del humedal.



En la Resolución 5.7 y la Resolución 8.14 se pidieron planes de manejo para todos los humedales RAMSAR. Con el apoyo y los fondos apropiados para su ejecución y para la capacitación del personal, lo que incluye un programa de vigilancia con indicadores sobre las características ecológicas del sitio.

Actualmente está en marcha el desarrollo de un plan específico para el PNAV, en el cual participan diferentes actores como el Ayuntamiento de Valencia, la Confederación Hidrográfica del Júcar y la Generalitat Valenciana.

Es en este punto donde una herramienta como la que se desarrolla en este TFM puede ser útil a la hora de gestionar de forma más sencilla el problema de eutrofización de la laguna de la Albufera.



2. Objetivos

El principal objetivo de este Trabajo Final de Máster [TFM] es la evaluación del estado trófico del agua de la laguna del PNAV aplicando modelos de Redes Neuronales Artificiales [RNA]. El deterioro de las aguas en la zona de estudio y la necesidad de revertir esta situación, justifica el objetivo de este TFM.

Se implementarán diferentes modelos de RNA para determinar la capacidad de estimación que tienen la aplicación de estas herramientas en espacios eutrofizados. Estos modelos estarán orientados a representar los principales procesos que tienen lugar en el crecimiento de las algas.

Con el fin de alcanzar el objetivo principal se van a llevar a cabo las siguientes actividades:

- 1) Recopilar información y revisar la bibliografía sobre el uso de RNA en el ámbito de los Recursos Hídricos [RRHH], más concretamente en la resolución de problemas relacionados con la calidad del agua.
- 2) Plantear una alternativa al uso de modelos convencionales para la modelación de los parámetros más habituales en la calidad del agua.
- 3) Realizar un análisis técnico de los resultados obtenidos tras el uso de RNA en la modelación de las concentraciones de clorofila-*a* en la laguna del PNAV.
- 4) Desarrollar una herramienta sencilla que aporte información sobre la dinámica del sistema y que pueda ser empleada por los responsables de la gestión del paraje facilitando la toma de decisiones.



3. Estado del Arte

En el presente apartado se realiza una recopilación de información en referencia al tema del TFM. Con el objetivo de ofrecer una idea general de cómo funcionan las RNA y el papel que han tenido dentro del ámbito de la modelación de RRHH, y más concretamente, en el estudio de la calidad de las aguas.

3.1 Introducción

Las RNA han llegado a ser extremadamente populares para la predicción y el pronóstico en un amplio número de áreas como la economía, la medicina, los RRHH etc. (Maier & Dandy, 2000). Aunque el concepto de neuronas artificiales fue introducido en 1943 (McCulloch & Pitts, 1943), los mayores avances en la investigación de las posibles aplicaciones de RNA florecieron desde la introducción del algoritmo de entrenamiento de *backpropagation* en 1986 (Rumelhart, Smolensky, McClelland, & Hinton, 1986).

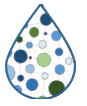
Entre los primeros modelos basados en RNA, cabe mencionar: Perceptron (1957), Adalain y Madeline (1960), Avalancha (1967), Retropropagación (1974), Hopfield y SOM (1980), ART (1986), etc.

Estos modelos estaban muy lejos de poder ser empleados por usuarios sin un elevado nivel de especialización o de ser aplicados para la resolución de problemas comunes. Actualmente, el enfoque ha cambiado y el uso de este tipo de técnicas está al alcance de cualquier usuario. Ya sea, mediante el uso de software de pago como MATLAB o software libres como R. En este sentido, la nueva situación permite desarrollar e investigar la utilidad de las redes en un ámbito cada vez más amplio.

3.1.1 Definición y descripción de RNAs:

En la literatura podemos encontrar un gran número de definiciones de RNA, algunas de ellas son:

- *“Redes neuronales artificiales son redes interconectadas masivamente en paralelo de elementos simples con organización jerárquica, las cuales intentan interactuar con objetos del mundo real del mismo modo que lo hace el sistema nervioso” (Kohonen, 1988).*



- *“(…) un sistema de computación constituido por un gran número de elementos simples de procesamiento muy interconectados, que procesan información por medio de su estado dinámico como respuesta a las entradas”* (Hecht-Nielsen, R., 1988).
- *“Una RNA es un sistema de procesamiento de información masivamente distribuido en paralelo que tiene ciertas características semejantes a las redes neuronales biológicas del cerebro humano”* (Haykin, 1994).
- *“Una nueva forma de computación, inspirada en modelos biológicos”* (Hilera & Martinez, 1995)
- *“Un modelo matemático compuesto por un gran número de elementos procesales organizados por niveles”* (Hilera & Martinez, 1995).
- *“Son sistemas adaptables que aprenden relaciones de un grupo de entradas y salidas y luego son capaces de predecir un conjunto de datos que no hayan sido vistos previamente, con características similares a los datos de entrada”* (Asce, 2000).
- *“Intento de implementar en un ordenador estructuras y funciones similares a las capacidades de procesamiento de datos del cerebro humano”* (Bravo-Oviedo & Kindermann, 2004).

Uno de los principales objetivos de los sistemas de aprendizaje autónomos es emular la habilidad que posee el ser humano para interactuar con el ambiente y aprender de dichas interacciones. Por ello, necesitan una estructura flexible, capaz de responder a incertidumbres y perturbaciones. Dentro de la teoría de Inteligencia Artificial, surgen las RNA como elementos capaces de proveer estructuras flexibles, mediante la integración con diversos sistemas de aprendizaje. Tales sistemas están formados por diferentes operaciones y pueden ser clasificados en dos tipos: aprendizaje supervisado y no supervisado, y dentro de este, aprendizaje auto-organizativo y aprendizaje por reforzamiento [(Jang & Sun, 1995), (Hoskins & Himmelblau, 1992),(Lin & Lee, 1996)].

En el presente TFM se ha centrado la atención en el uso de redes de aprendizaje supervisado. El cual consiste en entrenar la red con patrones, como valores de entrada, pertenecientes a una muestra poblacional representativa. Dicha muestra debe estar formada por parejas “patrón estímulos – respuesta observada” y debe de



ser elegida cuidadosamente. Como se expone en el apartado 3.1.2, al realizar el entrenamiento supervisado la respuesta de la red en cada patrón se compara con la respuesta observada, en virtud de esa comparación se reajustan los pesos sinápticos. El reajuste de los pesos sinápticos está orientado a alcanzar la respuesta observada con los valores de salida de la red.

Las redes podrían definirse como sistemas de aproximación universales en base a una serie de estímulos. Ya que, se ha ido demostrado que son capaces de aproximar funciones no lineales con gran precisión (Irie & Miyake, 1988);(Kurt Hornik, 1991). En contraste con los modelos estadísticos tradicionales de predicción, en las RNAs no es necesario elegir una función en concreto y es la misma red la que se adapta según las entradas y las salidas (Pujol Reig, 2009).

Las RNAs están compuestas por cuatro elementos básicos: las neuronas o nodos, las conexiones, los pesos sinápticos y las funciones de activación.

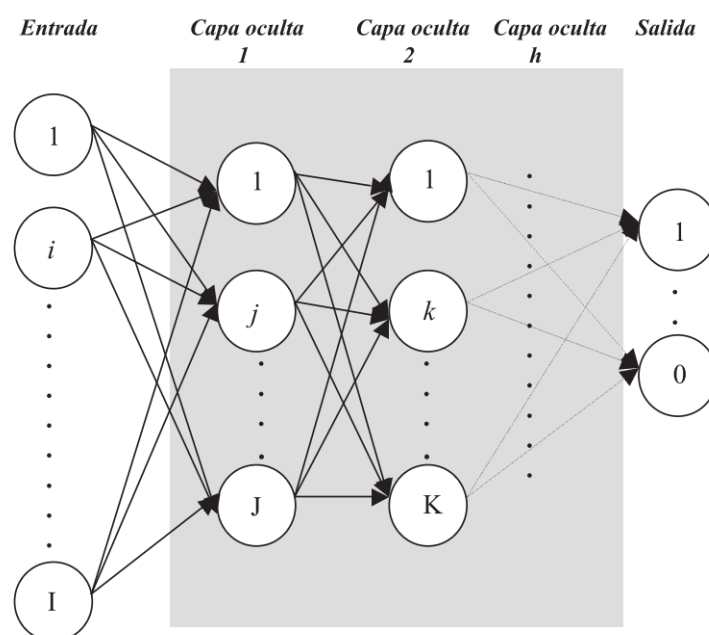
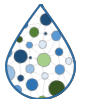


Ilustración 1 Estructura básica de una RNA (Bravo-Oviedo & Kindermann, 2004)

La distribución de neuronas dentro de la red se realiza formando niveles o capas, con un número determinado de dichas neuronas en cada una de ellas. A partir de su situación dentro de la red, se pueden distinguir tres tipos de capas:

- **De entrada:** es la capa que recibe directamente la información proveniente de las fuentes externas de la red.



- **Ocultas:** son internas a la red y no tienen contacto directo con el entorno exterior. El número de niveles ocultos puede estar entre cero y un número elevado. Las neuronas de las capas ocultas pueden estar interconectadas de distintas maneras, lo que determina, junto con su número, las distintas topologías de redes neuronales.
- **De salidas:** transfieren información de la red hacia el exterior.

La conectividad entre los nodos de una red neuronal está relacionada con la forma en que las salidas de las neuronas están canalizadas para convertirse en entradas de otras neuronas. La señal de salida de un nodo puede ser una entrada de otro elemento de proceso, o incluso ser una entrada de sí mismo (*conexión autorrecurrente*).

Cuando ninguna salida de las neuronas es entrada de neuronas del mismo nivel o de niveles precedentes, la red se describe como de conexión hacia delante. Cuando las salidas pueden ser conectadas como entradas de neuronas de niveles previos o del mismo nivel, incluyéndose ellas mismas, la red es de conexión hacia atrás (Matich, 2001).

Las reglas que rigen el funcionamiento y la estructura de una RNA son:

- I. El procesamiento de información se produce en muchos elementos simples llamados nodos, también denominados unidades, células o neuronas.
- II. Las señales se pasan entre nodos a través de enlaces de conexión.
- III. Cada enlace de conexión tiene un peso asociado que representa su fuerza de conexión.
- IV. Cada nodo aplica típicamente una transformación no lineal llamada una función de activación a su entrada de red para determinar su señal de salida.

El tipo de red empleada y su estructura, se define por: la arquitectura que representa la estructura de conexión entre los nodos, por su método de determinar los pesos de conexión y por la función de activación que emplea (Fausett, 1994).

3.1.2 Implementación de una red neuronal:

Una vez identificada la tipología más adecuada y la estructura que mejor se ajuste a las necesidades del usuario, la implementación de una RNA consta de dos fases.



En la primera parte, conocida como la *fase de entrenamiento*, se ajustan los pesos de las neuronas ocultas mediante la aplicación de un algoritmo de entrenamiento a fin de minimizar el error entre la respuesta de la red y el dato observado. Entre los algoritmos más populares empleados en este campo destaca el *backpropagation*.

Una vez alcanzada la optimización deseada en la fase de entrenamiento pasamos a la *fase de test o validación* en la cual se ejecuta el modelo y se comprueba el funcionamiento del modelo ante diferentes valores de entrada. Gracias a los ejemplos aprendidos en el entrenamiento la red deberá ser capaz de generalizar y dar respuestas correctas ante patrones de estímulos nuevos (Gestal, 2009).

El procedimiento en esta fase es similar al de la fase anterior, solo que tras cada respuesta de la red no se reajustan los pesos. Una de las claves para obtener buenos resultados en la fase de validación es entrenar con una muestra poblacional representativa. Si los datos de un problema se diferencian demasiado de los patrones de entrenamiento, la red tendrá dificultades para encontrar la respuesta correcta (Gestal, 2009).

3.2 Ventajas e Inconvenientes de las RNAs

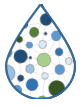
Como cualquier método empleado en la modelación de calidad de aguas, las RNA presentan ventajas e inconvenientes. Según la publicación realizada por Matich, (2001), entre las ventajas que ofrecen al usuario, destacan:

- **Aprendizaje adaptativo:**

La capacidad de aprendizaje adaptativo es una de las características más atractivas de redes neuronales. Esta se refiere a la capacidad de llevar acabo ciertas tareas mediante un entrenamiento con ejemplos ilustrativos.

Como las redes neuronales pueden aprender a diferenciar patrones mediante ejemplos y entrenamientos, no es necesario elaborar modelos a priori ni necesidad de especificar funciones de distribución de probabilidad.

Las redes neuronales son sistemas dinámicos autoadaptativos. Son adaptables debido a la capacidad de autoajuste de los elementos procesales (neuronas) que componen el sistema. Son dinámicos, pues son capaces de estar constantemente cambiando para adaptarse a las nuevas condiciones.



En el proceso de aprendizaje, los enlaces ponderados de las neuronas se ajustan de manera que se obtengan ciertos resultados específicos. Una red neuronal no necesita un algoritmo para resolver un problema, ya que ella puede generar su propia distribución de pesos en los enlaces mediante el aprendizaje. También existen redes que continúan aprendiendo a lo largo de su vida, después de completado su período de entrenamiento.

La función del diseñador es únicamente la obtención de la arquitectura apropiada. No es problema del diseñador el cómo la red aprenderá a discriminar. Sin embargo, sí es necesario que desarrolle un buen algoritmo de aprendizaje que le proporcione a la red la capacidad de discriminar, mediante un entrenamiento con patrones.

- **Sencillez:**

Si comparamos este tipo de modelos con los modelos mecanicistas clásicos, destaca la sencillez que los caracteriza tanto en su estructura como en su implementación.

- **Auto-organización:**

Las redes neuronales emplean su capacidad de aprendizaje adaptativo para auto-organizar la información que reciben durante el aprendizaje y/o la operación. Mientras que el aprendizaje es la modificación de cada elemento procesal, la auto-organización consiste en la modificación de la red neuronal completa para llevar a cabo un objetivo específico.

Cuando las redes neuronales se usan para reconocer ciertas clases de patrones, ellas auto-organizan la información usada. Por ejemplo, la red llamada *backpropagation* creará su propia representación característica, mediante la cual puede reconocer ciertos patrones.

Esta auto-organización provoca la generalización: facultad de las redes neuronales de responder apropiadamente cuando se les presentan datos o situaciones a las que no había sido expuesta anteriormente. El sistema puede generalizar la entrada para obtener una respuesta. Esta característica es muy importante cuando se tiene que solucionar problemas en los cuales la



información de entrada no es muy clara; además permite que el sistema dé una solución, incluso cuando la información de entrada está especificada de forma incompleta

- **Tolerancia a fallos:**

Las redes neuronales fueron los primeros métodos computacionales con la capacidad inherente de tolerancia a fallos. Comparados con los sistemas computacionales tradicionales, los cuales pierden su funcionalidad cuando sufren un pequeño error de memoria. En las redes neuronales, si se produce un fallo en un número no muy grande de neuronas y aunque el comportamiento del sistema se ve influenciado, no sufre una caída repentina.

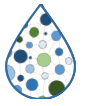
Hay dos aspectos distintos respecto a la tolerancia a fallos:

- a) Las redes pueden aprender a reconocer patrones con ruido, distorsionados o incompletos. Esta es una tolerancia a fallos respecto a los datos.
- b) Las redes pueden seguir realizando su función (con cierta degradación) aunque se destruya parte de la red.

La razón por la que las redes neuronales son tolerantes a los fallos es que tienen su información distribuida en las conexiones entre neuronas, existiendo cierto grado de redundancia en este tipo de almacenamiento. La mayoría de los ordenadores algorítmicos y sistemas de recuperación de datos almacenan cada pieza de información en un espacio único, localizado y direccionable. En cambio, las redes neuronales almacenan información no localizada. Por lo tanto, la mayoría de las interconexiones entre los nodos de la red tendrán sus valores en función de los estímulos recibidos, y se generará un patrón de salida que represente la información almacenada.

- **Capacidad de generalización:**

Ante la entrada de datos nuevos es capaz de producir resultados coherentes de acuerdo con la naturaleza del problema para el cual han sido entrenadas.



- **Operación en tiempo real:**

Una de las mayores prioridades, casi en la totalidad de las áreas de aplicación, es la necesidad de realizar procesos con datos de forma muy rápida. Las redes neuronales se adaptan bien a esto debido a su implementación paralela. Para que la mayoría de las redes puedan operar en un entorno de tiempo real, la necesidad de cambio en los pesos de las conexiones o entrenamiento es mínimo.

- **Fácil inserción dentro de la tecnología existente:**

Una red individual puede ser entrenada para desarrollar una única y bien definida tarea (tareas complejas, que hagan múltiples selecciones de patrones, requerirán sistemas de redes interconectadas). Con las herramientas computacionales existentes, una red puede ser rápidamente entrenada, comprobada, verificada y trasladada a una implementación hardware de bajo coste. Por lo tanto, no se presentan dificultades para la inserción de redes neuronales en aplicaciones específicas, por ejemplo de control, dentro de los sistemas existentes. De esta manera, las redes neuronales se pueden utilizar para mejorar sistemas en forma incremental y cada paso puede ser evaluado antes de acometer un desarrollo más amplio.

Pero los sistemas neuronales no están exentos de ciertos inconvenientes. Uno por el que fueron muy impopulares en su momento, es que habitualmente realizan un complejo procesamiento que supone millones de operaciones. Por lo que no es posible seguir paso a paso el razonamiento que les ha llevado a extraer sus conclusiones. Por ello han sido considerados modelos de caja negra por un gran número de autores. Sin embargo, en redes pequeñas, mediante simulación o por el estudio de los pesos sinápticos sí es posible saber, al menos, que variables de las introducidas han sido más relevantes para alcanzar el resultado deseado (Gestal, 2009).

La elección de una estructura adecuada es primordial para alcanzar el rendimiento deseado en una RNA. Definida por el número de capas y de neuronas, una estructura

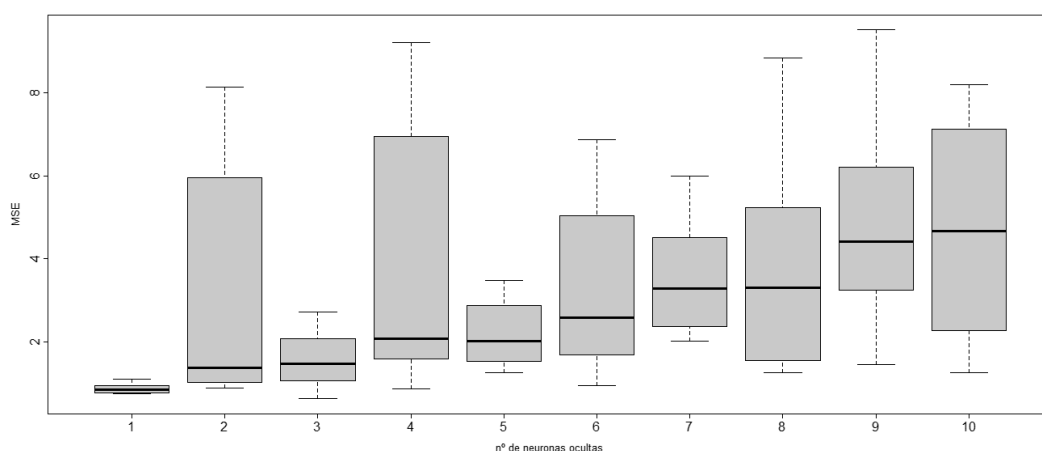


que no se ajuste a las necesidades de la red frente al problema planteado puede fomentar la aparición de un fenómeno conocido como sobreentrenamiento.

- **Sobreentrenamiento:**

El número de neuronas en la capa oculta proporciona a la RNA la capacidad de aproximar las relaciones no lineales existentes entre los datos. Éste debe ser seleccionado con cuidado puesto que un número excesivo de neuronas provoca demasiada flexibilidad al modelo que puede traducirse en el fenómeno de pérdida de la capacidad de predicción de la red conocido como sobreentrenamiento (Salazar, Rodríguez, & Cabrera-ríos, 2006).

A continuación, se muestra un gráfico diagrama de caja, que refleja un claro caso de sobreentrenamiento. Si se observa, a medida que se va aumentando el número de neuronas ocultas de la red, se obtiene un error cuadrático medio mayor o lo que es lo mismo, peores resultados.



Gráfica 1 Error Cuadrático Medio de una RNA variando el número de neuronas oculta. Fuente: Elaboración propia

Pero las desventajas reales o los problemas más comunes en el uso de RNA para modelar, residen en la fase de entrenamiento. Ya que las respuestas pobres o con poca precisión de una red suele ser consecuencia de ejecutar mal dicha fase. Generalmente los motivos son:



- **Patrones insuficientes:** si el número de muestras empleadas para el entrenamiento no es suficientemente grande puede que la red no sea capaz de generalizar cuando se enfrente a la fase de validación.
- **Muestras no representativas:** otro aspecto importante es disponer a la red de patrones heterogéneos y que representen, en lo medida de lo posible, todos los escenarios con los que se puede enfrentar el modelo. Si no se cubre dicha necesidad, puede que la red no sepa responder correctamente ante situaciones para las que no ha sido entrenada

3.3 RNA y Recursos Hídricos

La predicción de variables hidroclimatológicas como la precipitación y los caudales se ha convertido en una necesidad imperativa en los procesos de gestión de Sistemas Hídricos. El nacimiento de la nueva disciplina que incluye el uso de las RNA se ha ido aplicando a diferentes estudios de aprovechamiento y conservación de RRHH (Obregón, Fragala, & Prada, 2003).

Desde los años noventa se ha ido aplicando este nuevo enfoque en el campo de los RRHH. Los trabajos desarrollados han demostrado la utilidad de incluir el uso de técnicas de inteligencia artificial en la simulación y modelación de los fenómenos físicos, químicos y biológicos del agua. Se obtienen buenos resultados por el comportamiento no lineal de las variables (Ali Najah, Elshafie, Karim, Jaffar, & Lecturer, 2009).

Se pueden encontrar numerosos estudios centrados en validar el uso de las redes neuronales en la resolución de problemas relacionados con la hidrología superficial como la Tesis Doctoral de Pujol Reig (2009). En la cual se desarrolla un software específico para aplicar estas técnicas en la predicción de caudales.

Un trabajo que demuestra la utilidad de las RNA en este campo, es el realizado por J. C. Ochoa-Rivera, Andreu, & García-Bartual, (2007). Donde se analiza la influencia de varios parámetros hidrológicos en la modelación de Sistemas de Recursos Hídricos [SSRRHH] en el sistema Júcar. Para ello, se plantearon tres simulaciones mediante el uso de series de caudales sintéticas generadas con tres modelos estocásticos: modelo autorregresivo integrado de promedio móvil [ARMA]; modelo de desagregación temporal de Lane, (1979) y un modelo no lineal basado en una red



neuronal del tipo *Multilayer Perceptron* [MLP] con una componente aleatoria incorporada. La validación de los modelos se realizó comparando estadísticamente las sequias de las series sintéticas con las observadas en registros históricos. Se emplearon cinco series de caudales históricos tomados por la Confederación Hidrográfica del Júcar durante el periodo 1940-2000. Las simulaciones de gestión del SRRHH se ejecutaron con el sistema de soporte a la decisión AQUATOOL (Andreu, Capilla, & Sanchís, 1996) bajo un enfoque probabilístico con el módulo SIMRISK. Este enfoque da probabilidades de fallos de demanda de los SRRHH, se utilizó para evaluar la influencia de los tres modelos estocásticos aplicados en los resultados de la simulación. Tras la ejecución de las tres simulaciones, se observó que con el modelo ARMA se obtenía el menor número de fallos por lo que infravaloraba los déficits sufridos por el sistema. En cambio, el modelo de desagregación y el MLP daban como resultado un número de fallos similar. Pero, aunque los dos últimos modelos tenían resultados similares, el MLP precisaba cuatro veces menos número de parámetros, por lo que era más acorde con el principio de parsimonia.

Otro ejemplo similar, es el artículo publicado por Juan Camilo Ochoa-Rivera, (2008). En el cual se usan RNA para desarrollar un modelo con el objetivo de predecir futuros escenarios de sequía. Se aporta información sobre la duración, la intensidad, la magnitud y la frecuencia en la que se dan dichos escenarios. Los resultados se compararon con la respuesta de uno de los modelos más empleados en la resolución de este tipo de problemas, el ARMA. Finalmente, se concluyó que como en el caso comentado anteriormente, se obtenían mejores resultados con la RNA.

3.4 RNA y Calidad de las aguas

La modificación no natural del estado trófico de las masas de agua provoca grandes pérdidas ecológicas. A continuación, se comentan algunos de trabajos realizados que, dentro de la literatura consultada, guardan más relación con la temática de este TFM. En primer lugar, se centra la atención en comprobar si este tipo de técnicas obtiene buenos resultados en la modelación de la calidad del agua, como ya se había demostrado en otros ámbitos de los RRHH.



Un trabajo que se centra en validar las redes en este ámbito es el publicado por Abyaneh, (2014). En él se comparan los resultados obtenidos por un modelo basado en una Regresión Lineal Multivariante [RLM] y un modelo basado en una RNA. El principal objetivo del trabajo fue modelar dos de las variables más importantes en la operatividad de una Estación Depuradora de Aguas Residuales [EDAR]. Dichos parámetros fueron: la demanda biológica de oxígeno [DBO] y la demanda química de oxígeno [DQO]. Los *inputs*

empleados en la predicción de las variables objetivo, fueron: temperatura [T], pH, sólidos suspendidos totales [SST] y los sólidos totales [ST]. La evaluación de los modelos se realizó a partir del coeficiente de correlación [r] y el error cuadrático medio [RMSE]. Finalmente, los resultados demostraron que los resultados de la RNA se aproximaban más a los observados, que los del modelo RLM. Siendo el r de la red superior a 0.8 para ambas variables, casi el doble que lo conseguido aplicando el modelo MLR.

Otro ejemplo en el que se estudia el uso de RNA para la modelación de la calidad de aguas naturales superficiales de tipo río, es el publicado por Gazzaz, Yusoff, Aris, Juahir, & Ramli, (2012). El trabajo se centra en la simulación del índice de calidad de agua [WQI] en el río Kinta, Malasia. Dicho índice fue adoptado en 1974 por el departamento de Medio Ambiente de Malasia y su valor final depende de las concentraciones de las siguientes variables: DO, DBO, DQO, pH, NH₃-H y SST. Su cálculo se caracteriza por ser un proceso largo y costoso, de ahí que se deseara encontrar un modelo con la capacidad predicción. Tras el pretratamiento de la base de datos con la que se deseaba entrenar, validar la red, se realizó la optimización de la arquitectura y la estructura de la RNA. Las predicciones de WQI de la red presentaron una correlación significativa, positiva y muy alta ($r = 0.977$) con los valores de WQI observados. Lo que implica que las predicciones del modelo explican alrededor del 95.4% de la variación en los valores de WQI medidos. El enfoque de este artículo demuestra la utilidad de estas nuevas herramientas y la capacidad de adaptación que poseen para resolver problemas complejos. No solo para simular variables concretas sino también para modelar índices de calidad que indiquen el estado ecológico de aguas naturales, ahorrando mucho trabajo a los agentes encargados de su evaluación.

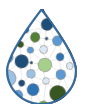


A. Najah, El-Shafie, Karim, & El-Shafie, (2013), plantean la resolución de un problema ambiental asociado a la gran contaminación sufrida por del rio Johor, Malasia, mediante el entrenamiento y validación de dos RNAs. Una del tipo MLP y otra del tipo *radial basis function neural networks* [RBF-NN]. Además, las respuestas de ambas redes fueron comparadas con los resultados obtenidos por un modelo de RLM. Las variables de interés fueron: la turbidez, la concentración de solidos suspendidos y la conductividad eléctrica. Los resultados obtenidos demuestran que es difícil encontrar un modelo convencional que represente los parámetros de calidad estudiados. Debido principalmente, a las relaciones no lineales entre ellos y a la naturaleza estocástica de los procesos químicos. La variable de la que se obtuvo peores resultados fue la turbidez, principalmente por su dependencia directa con parámetros no incluidos en el estudio como: la concentración de algas, el caudal o las lluvias. Finalmente, se demuestra mediante el cálculo de los coeficientes de eficiencia y el error en la predicción, como ambas redes resuelven mejor el problema planteado con errores promedios y tiempos computacionales menores que el MLP.

Una vez demostrada la capacidad de estas nuevas técnicas para ajustarse y reproducir con buenos resultados, fenómenos o aspectos relacionados con la calidad de las aguas. Se concreta más y se centra la atención en comprobar si las RNA ya han sido empleadas para resolver problemas relacionados con la eutrofización de las aguas y si los resultados han sido positivos, como en los casos anteriores.

Cabe destacar, que los problemas de eutrofización no afectan solo a aguas naturales, sino también a reservas de agua no naturales, como los embalses. La concentración alta de nutrientes en embalses suele estar originada por contaminación difusa o localizada que puede provocar la aparición de *blooms* de microalgas. En estos casos se reduce la capacidad del recurso hídrico y produce importantes pérdidas, a nivel económico y social, a las poblaciones dependientes.

Por todo ello muchos de los esfuerzos por parte de los gestores se centran en controlar el estado trófico de las masas de agua. A continuación, se describen artículos centrados en la modelación de las dinámicas poblacionales de microalgas o de compuestos ligados ellas.



En primer lugar, nombrar el estudio realizado por Cho, (2014). En el cual se trató de identificar los factores que afectaban a la aparición de *algal blooms* y se desarrolló un modelo mediante el uso de RNA. El objetivo del trabajo era predecir las variaciones en la concentración de clorofila-*a* en el embalse de *Lake Juam*, situado en Corea del Sur. La motivación de este estudio residía en evitar los problemas asociados a dichos *blooms*, ya que eran los responsables del mal olor y sabor al agua potable, de la coloración de las aguas, la destrucción de los ecosistemas, la colmatación de los filtros en las potabilizadoras y la toxicidad por las poblaciones de cianobacterias. Para el desarrollo de los modelos se dispusieron las bases de los datos registrados de calidad, de hidrología y de meteorología, en el periodo 2008-2010. A fin de identificar los factores más influyentes sobre dinámica de las comunidades de algas, se plantearon nueve escenarios en los cuales se variaba el número de *inputs*. A continuación, se muestran las características de cada escenario simulado y los resultados obtenidos. Cabe destacar, que los datos empleados como *inputs* han sido tomados en (t-1), por lo que la predicción se realiza con un día de antelación.

Tabla 1 Escenarios propuestos (Cho, 2014)

Escenarios	Inputs usando un análisis de correlación
Escenario 1	Chl-a, TOC, pH, temperatura atmosférica, temperatura del agua, nivel del agua, duración de las horas de sol, caudal de salida, conductividad y caudales de entrada.
Escenario 2	Chl-a, TOC, pH, temperatura atmosférica, temperatura del agua, nivel del agua, duración de las horas de sol, caudal de salida y conductividad.
Escenario 3	Chl-a, TOC, pH, temperatura atmosférica, temperatura del agua, nivel del agua, duración de las horas de sol y caudal de salida.
Escenario 4	Chl-a, TOC, pH, temperatura atmosférica, temperatura del agua, nivel del agua y duración de las horas de sol.
Escenario 5	Chl-a, TOC, pH, temperatura atmosférica, temperatura del agua y nivel del agua.
Escenario 6	Chl-a, TOC, pH, temperatura atmosférica y temperatura del agua.
Escenario 7	Chl-a, TOC, pH y temperatura atmosférica.
Escenario 8	Chl-a, TOC y pH.
Escenario 9	Chl-a y TOC.

Todos los datos se toman con un día de antelación.



Tabla 2 r^2 , RMSE, NSE y % de Diferencia entre las predicciones y los datos observados de clorofila-*a* (Cho, 2014)

Índices estadísticos	Escenarios								
	1	2	3	4	5	6	7	8	9
r^2	0.37	0.02	0.37	0.31	0.18	0.81	0.81	0.83	0.85
RMSE (mg/m ³)	5.54	8.19	8.70	13.29	7.30	4.99	4.86	4.96	4.88
NSE	-0.08	-3.70	-1.90	-9.56	-1.91	0.80	0.79	0.82	0.85
%Diferencia	9.50	9.42	43.35	120.6	22.5	0.92	4.56	1.51	0.46

Como se observa en la tabla anterior, la red que mejor responde es la generada para el escenario nueve. Con ello se demuestra que no siempre es necesario desarrollar complejos modelos para obtener buenos resultados. En ocasiones, el incluir gran número de *inputs* puede introducir un ruido innecesario reduciendo la eficiencia del modelo.

Otro ejemplo del uso de redes para la modelación de las fluctuaciones de clorofila-*a* de un ecosistema acuático es el publicado por Guallar, Delgado, Diogène, & Fernández-Tejedor, (2016). En el cual mediante el uso de redes neuronales artificiales trataron de predecir la ausencia o presencia y la abundancia de dos familias de microalgas en la *Badia dels Alfacs* localizada en la parte meridional del delta del Ebro. Para el desarrollo del estudio se emplearon datos de: calidad de aguas, salinidad, temperatura, oxígeno, clasificaciones de las poblaciones de fitoplancton, etc. Se desarrollaron dieciséis modelos independientes, ocho para cada especie y se varió el tiempo de predicción entre una y dos semanas de antelación. Tras un primer análisis de sensibilidad para identificar las variables más significativas que afectan a cada una de las especies analizadas, se implementó las RNA. Los mejores resultados de r^2 para cada especie se situaba entre 0.82-0.70 en la fase de entrenamiento y 0.82-0.68 en la fase de validación. La aproximación alcanzada en la simulación del crecimiento de ambas especies, demuestra la elevada capacidad de las RNA. Aunque la mayoría de estudios sobre el uso de estas técnicas para simulación se centran en conocer la concentración de clorofila-*a*, en este caso se demuestra que es posible representar la dinámica poblacional de una especie concreta, con buenos resultados.



Conocer las variaciones de clorofila-*a* en un ecosistema acuático nos da una idea del crecimiento total de la biomasa algal. Dentro de las familias de microalgas que componen dicha biomasa, encontramos las cianobacterias o algas azules que en ocasiones pueden liberar toxinas en ciertas etapas de su vida. La liberación aguada de estos compuestos puede producir la intoxicación de la fauna del paraje, especialmente de la ictiofauna del ecosistema afectado. Los animales afectados por esta toxina pueden llegar a morir por hemorragia o falla hepática. De ahí que en el estudio realizado por Heddam, (2016), se centren en modelar mediante RNA el pigmento más característico de estas microalgas conocido como ficocianina, en vez de la clorofila-*a*. El objetivo de ese trabajo es conocer la relación entre la concentración de ficocianina y la dinámica poblacional de las cianobacterias, mediante el uso de RNA del tipo MLP. El caso de estudio del proyecto se localiza en el río *Charles*, el cual desemboca en el puerto de Boston. Los parámetros usados en el desarrollo del modelo fueron: la temperatura del agua [T], el oxígeno disuelto [OD], pH, la conductividad específica [CE] y la ficocianina [PC]. Los datos procedían de un programa de monitoreo mediante sondas de la EPA (*The US Environmental Protection Agency*). Los cuatro primeros parámetros se emplearon como *inputs* y el último de ellos como *output*. Además, los resultados del MLP se compararon con los obtenidos mediante una RLM. Tras el planteamiento de once escenarios en los cuales se variaba los parámetros empleados como *inputs* en la red se concluyó, que la red más eficaz era la que empleaba los cuatro parámetros (T, OD, pH y CE). Se plantearon diferentes estructuras internas, variando el número de neuronas ocultas, alcanzando la mayor optimización del modelo con 13 neuronas ocultas. Finalmente, se consiguieron coeficientes de correlación para la predicción de la ficocianina de 0.976 en la fase de entrenamiento, 0.976 en la fase de validación y 0.975 para la fase de validación. Además, se obtuvieron mejores resultados que mediante el uso de MLR, y se determinó que el *input* con más peso o con más influencia sobre el *output* era el pH.

Por último, para concluir el presente apartado, cabe mencionar el estudio realizado por Lu, Chen, Liu, & Shao, (2016). Trabajo que ha servido como referente para la realización del presente TFM. En él se desarrolla un modelo basado en RNA, con el objetivo de modelar la concentración de clorofila-*a* en el lago de Champlain. La



muestra de datos usada en las fases de entrenamiento y validación, procedían de una campaña de monitoreo de la calidad del agua que duró 21 años (1992-2012). Al igual que en los casos anteriores se realizó un primer análisis para identificar los parámetros que influían más sobre la concentración de clorofila-*a*. Los primeros resultados demostraron que las variables más influyentes eran: el nitrógeno total, la turbidez y el nitrógeno total.

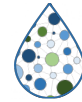
En la tabla 3 se muestran los 7 escenarios planteados junto con la estructura de la red en cada caso, los coeficientes de determinación [r^2] y el ratio de desviación estándar [RSR].

Tabla 3 Resultados estimación de clorofila-*a* dependiendo los inputs (Lu, Chen, Liu, & Shao, 2016).

<i>Inputs</i>	Estructura neuronal	Muestra entrenamiento		Muestra de Validación	
TP, TN, SD, DP, pH, T, DO	7-12-1	0.82	0.62	0.81	0.53
TP, TN, SD, DP, pH, T	6-11-1	0.82	0.83	0.76	1.08
TP, TN, SD, DP, pH	5-11-1	0.79	1.10	0.74	1.25
TP, TN, SD, DP	4-7-1	0.76	1.49	0.73	1.45
TP, TN, SD	3-5-1	0.81	0.85	0.79	0.83
TP, TN	2-4-1	0.75	1.39	0.73	1.32
TP	1-3-1	0.76	1.26	0.72	1.53

A diferencia de lo indicado por otros autores como Cho, en el presente estudio se observa como la red con mayor número de *inputs* es la que tiene mejor respuesta. Además, se comprueba que la variación de datos observados es muy alta principalmente, porque las presiones a las que estaba sometido el lago en 1992 son diferentes a las que se den 2012. Aun así, los resultados se consideran satisfactorios.

Mediante los artículos comentados, queda demostrada la utilidad de las RNA para la resolución de problemas relacionados con la eutrofización de las aguas.



4. Metodología

En este apartado de la memoria, se realiza una descripción detallada de la metodología empleada en el desarrollo de este TFM. Para ello se definen todas las características propias de las RNAs que han sido modificadas con la finalidad de optimizar el modelo en la medida de lo posible.

Además, se describe las fases previas al diseño de la red como puede ser el pretratamiento de datos. Siendo dicha fase imprescindible para alcanzar posteriormente un correcto entrenamiento del modelo.

En el siguiente esquema se resume el procedimiento seguido para el desarrollo de cada una de las RNAs diseñadas. En cada una de las fases se especifican los apartados donde se describen los pasos más importantes.

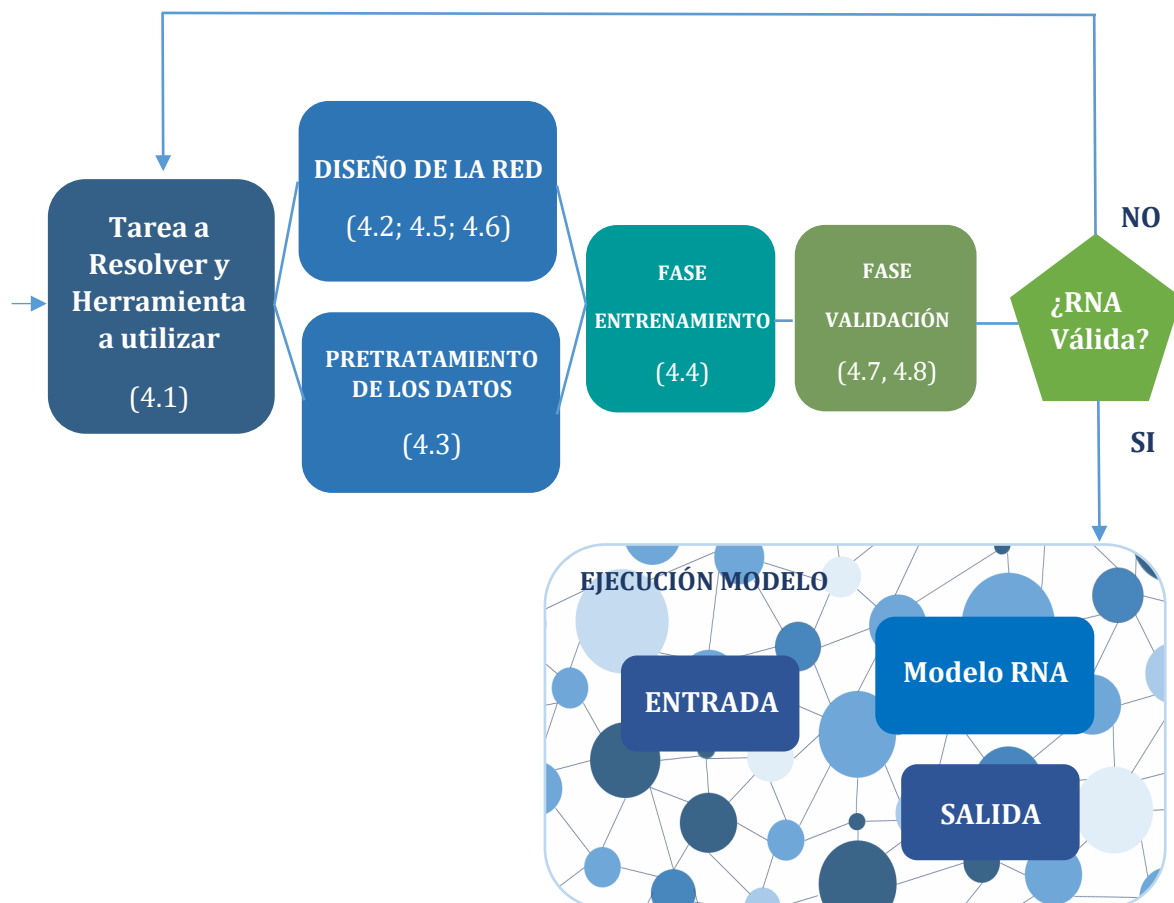


Ilustración 2 Esquema de la metodología para la construcción de una RNA : (Gestal, 2009)



4.1 Software

Actualmente existe un gran número de softwares especializados o con módulos específicos en el uso de RNAs. Algunos de ellos ya están desarrollados y entrenados para resolver problemas concretos. Otros, con mayor versatilidad para que el usuario pueda modificar la red acorde a sus necesidades. Entre los softwares para la implementación de RNAs más empleados destacan:

- **Neural Network Toolbox:** es un módulo del software matemático de pago MATLAB. Ofrece una amplia variedad de arquitecturas y funciones de entrenamiento para modelizar sistemas no lineales de un modo sencillo mediante RNA. Las *Apps* disponibles en *Neural Network Toolbox* permiten diseñar, entrenar, visualizar y simular interactivamente la red para posteriormente generar el código MATLAB equivalente y así poder automatizar el proceso. Además, soporta aprendizaje supervisado, no supervisado y reforzado mediante el uso de arquitecturas como: Perceptrón, perceptrón multicapa, Adaline, base radial, mapas auto-organizados, redes competitivas, etc.
- **Paquetes específicos de R:** es un software para el análisis estadístico libre. Entre las operaciones que ofrece destaca: almacenamiento y manipulación efectiva de datos, en particular matrices. Además de una amplia, coherente e integrada colección de herramientas para análisis de datos, posibilidades gráficas para análisis de datos, que funcionan directamente sobre pantalla o impresora. y un lenguaje de programación bien desarrollado, simple y efectivo, que incluye condicionales, ciclos, funciones recursivas y posibilidad de entradas y salidas (Team, 2000). Entre las herramientas que dispone, podemos encontrar paquetes creados específicamente para la implementación de RNA.

Para el desarrollo del presente TFM se ha usado el software R. Siendo la mejor opción entre los descritos, por tratarse de un software libre y por ofrecer la posibilidad de programar un código que cubra las necesidades de proyecto. Dentro de dicho software existen gran cantidad de librerías para la implementación de RNAs.



Con el fin de identificar la librería de R, que mejor se ajuste al proyecto, se han realizado unas pruebas preliminares para conocer los pros y los contras de cada una.

Se concluye que el paquete **Neuralnet** ofrece una gran versatilidad a la hora de ajustar la red a los requerimientos del estudio. La principal limitación del presente paquete aparece cuando se desea diseñar redes muy concretas como las recurrentes [RNR] o los mapas de Kohonen, siendo imposible su implementación.

Para suplir la carencia del paquete Neuralnet se ha decidido desarrollar algunas redes con el paquete RSNNS. El paquete **RSNNS** es una biblioteca específica creada para la implementación de tipologías de RNA muy concretas. Este paquete permite usar el software SNNS en R. El SNNS (*Stuttgart Neural Network Simulator*) es un software específico para simular redes neuronales en Unix workstations, desarrollado en el IPVR (*Institute for Parallel and Distributed High Performance Systems*) de la Universidad de Stuttgart. El objetivo del SNNS era crear una plataforma eficiente y flexible que facilite las investigaciones para la aplicación de redes neuronales. Como se desea comparar resultados entre redes del tipo MLP y RNN, se combinará el uso del paquete Neuralnet y el RSNNS en las ocasiones que se requiera.

4.2 Tipo de redes

Existe un gran número de tipologías de RNA, pudiendo clasificarse por la estructura de la red, el tipo de entrenamiento o por el algoritmo que emplean. Una de las clasificaciones más empleadas se centra en la diferenciación por el patrón de conexiones que presenta. Así, se definen dos tipos básicos de redes.

- **Redes de propagación hacia delante** (*feed-forward*). Es la forma más básica de red. Las neuronas de cada nivel sólo están conectadas con las neuronas de los niveles posteriores por lo que la información se propaga hacia delante. Algunas de las redes de este tipo más conocidas son: Perceptron, Adaline (ADAPTative LINear Elements), Madaline6 (Multiple Adaline) y el MLP. Todas ellas suelen aplicarse a la aproximación de funciones o al reconocimiento o clasificación de patrones (Santín González & Valiño Castro, 2003).



- **Redes de propagación hacia atrás o recurrentes** (*feedback*). En estas redes, las neuronas pueden estar conectadas indistintamente con neuronas de niveles previos, posteriores, de su mismo nivel (conexiones laterales) o incluso con ellas mismas (autorrecurrentes),(Fausett, 1994).

Con la intención de alcanzar los objetivos propuestos, se ha planteado el modelo mediante el uso de dos tipos de redes, una de propagación hacia delante y otra recurrente:

- **Multilayer Perceptron [MLP]**: son las redes más empleadas y están caracterizadas por constar de unidades que se agrupan en tres tipos de capas: la capa de entrada asimilable a las variables independientes, la capa de salida que corresponde a la variable dependiente y una serie de capas intermedias denominadas capas ocultas. Cada unidad de una capa está unida a otra mediante conexiones, que pueden ser hacia delante, hacia atrás o mixtas (Hilera & Martinez, 1995).
- **Jordan Neural Network**: También conocida como red recurrente simple o red parcialmente recurrente. Se puede estudiar como una red de propagación directa con neuronas adicionales de contexto en la capa de entrada. Estas neuronas de contexto toman la entrada a partir de sí mismas (conexión de retorno directa, *direct feedback*) y de las neuronas de salida. Las neuronas de contexto preservan el estado actual de la red. En la red de *Jordan*, la cantidad de neuronas de contexto y de salida debe ser la misma. Las conexiones recurrentes que van desde el *output* hasta las neuronas de entrada llevan asociado un parámetro “m”, que generalmente toma un valor constante y positivo menor a 1 (Jordan, 1986). Algunos autores recomiendan el uso de esta tipología de red a la hora de representar series temporales, como es en el caso de estudio.

El objetivo de probar el funcionamiento del modelo mediante el uso de una red más específica como la *Jordan*, es introducir en el modelo la relación entre el crecimiento poblacional del fitoplancton con o sin competencia intraespecífica. En el caso de las microalgas, aun existiendo disponibilidad ilimitada de nutrientes la población llega a un punto donde el tamaño de la población supera la capacidad de carga del



ecosistema. Es decir, el crecimiento no es exponencial, se ajusta una función logística.

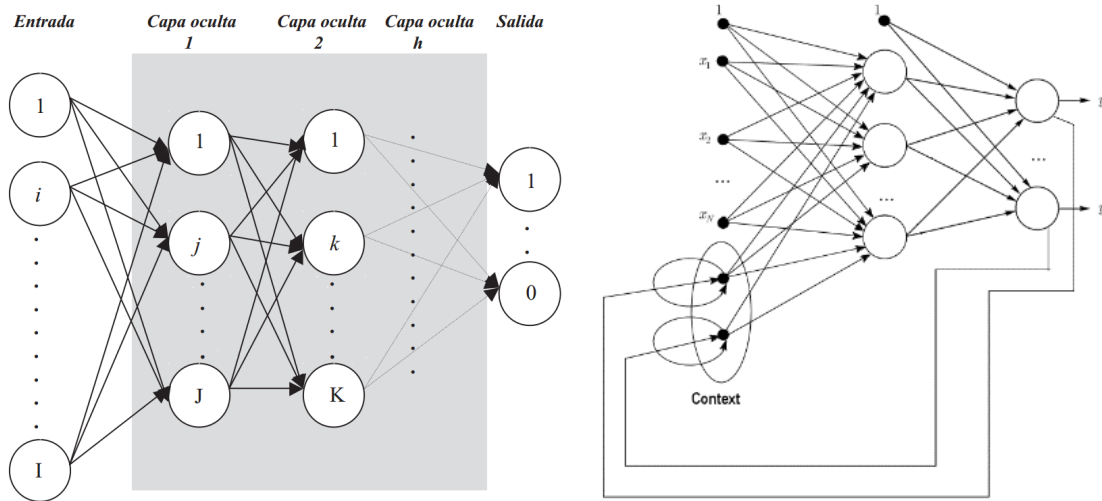


Ilustración 3 Topología de red neuronal con conexiones hacia delante (Bravo-Oviedo & Kindermann, 2004)

4.3 Pretratamiento de los datos

Antes de empezar con el diseño de la red, se ha realizado un primer análisis general de las muestras de datos que van a ser empleadas. Para ello se ha calculado el coeficiente de correlación de Pearson de algunas de las variables relacionadas directa o indirectamente con el crecimiento de las microalgas. Dicho coeficiente se ha empleado para hacer una primera criba entre todas las variables que podían tener alguna relación con la variable objetivo.

- El Coeficiente de correlación de Pearson [r] entre dos variables se define como la variancia de las dos variables, dividido por el producto de sus desviaciones estándar.

$$r_{xy} = \frac{\sum x_i y_i - nXY}{nS_x S_y} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2} \sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

Donde:

r = coeficiente de correlación de Pearson.

$\sum x_i y_i$ = sumatoria de los productos de ambas variables.

$\sum x$ = sumatoria de los valores de la variable independiente.



Σy = sumatoria de los valores de la variable dependiente.

Σx_i^2 = sumatoria de los valores al cuadrado de la variable independiente.

Σy_i^2 = sumatoria de los valores al cuadrado de la variable dependiente.

n = tamaño de la muestra en función de parejas.

Una vez seleccionar el conjunto de variables de entrada con las que se va a trabajar, se acotan las muestras de entrenamiento y validación. Es primordial que los datos sean representativos del problema que se desea abordar, es decir que está reflejado en ellos. Si no es así, la red será incapaz de responder correctamente.

A continuación, se deben detectar errores o patrones que pudiesen introducir ruido durante el entrenamiento de la red. La depuración de datos es especialmente relevante cuando se trabaja con muestras experimentales, donde son más comunes los errores instrumentales o humanos durante las mediciones.

Tras la depuración, se realiza un primer análisis de los datos mediante la evaluación de características estadísticas de la muestra como el máximo, el mínimo, la desviación estándar etc. Además, a fin de reducir posibles errores se normalizan los datos.

La normalización de los datos se puede realizar en el intervalo [0,1] o el intervalo [-1,1]. El objetivo de este proceso es asegurar que todas las variables (entradas a la RNA) reciben igual atención en el proceso de entrenamiento y están escalados de acuerdo a los límites de las funciones de activación utilizadas en las neuronas de la capa de salida de la red (Gestal, 2009).

Seguidamente se muestra la fórmula introducida en el código para automatizar la normalización de los datos de entrada y la desnormalización de los datos de salida.

$$x_{norm} = \frac{(x * (0.9 - 0.1)) + (\max(Data) * 0.1 - \min(Data) * 0.9)}{\max(Data) - \min(Data)}$$

$$x_{desnorm} = \frac{(x * ((\max(Data) - \min(Data)))) - (\max(Data) * 0.1 - \min(Data) * 0.9)}{0.9 - 0.1}$$



Al no encontrar diferencias significantes en los resultados al variar las funciones, se ha decidido emplear la función de activación logística en todas las RNAs. Por ello, la normalización se realizó en el intervalo $[0,1]$ con una pequeña modificación: los límites se acotaron a $[0.1,0.9]$. Con dicha modificación se reducen las posibilidades de que durante la fase de ejecución algún dato se salga del rango para el cual está entrenada la red. Por último, para evitar que los valores se saliesen del rango indicado, se normalizo empleando el máximo y mínimo de toda la muestra de datos [Data].

4.4 Algoritmos de entrenamiento

El algoritmo de entrenamiento se emplea para localizar los óptimos y ajustar el peso de las neuronas de forma que el modelo responda lo más eficientemente posible. El paquete Neuralnet, da la opción de elegir el algoritmo que se desea emplear para el entrenamiento. A continuación, se muestra una pequeña descripción de cada uno de ellos.

- **“Backprop”**: El funcionamiento de la red *backpropagation* consiste en el aprendizaje mediante un conjunto predefinido de pares de entradas-salidas. Primero se aplica un patrón de entrada como estímulo para la primera capa de las neuronas de la red y se va propagando a través de todas las capas superiores hasta generar una salida. Finalmente se compara el resultado en las neuronas de salida con la respuesta que se desea obtener y se calcula un valor de error para cada neurona de salida. A continuación, estos errores se transmiten hacia atrás, partiendo de la capa de salida hacia todas las neuronas de la capa intermedia que contribuyan directamente a la salida. Este proceso se repite, capa por capa, hasta que todas las neuronas de la red hayan recibido un error que describa su aportación relativa al error total.
- **“Rprop+ y Rprop-”**: Métodos de aprendizaje basados en el *backpropagation* pero que incluyen una nueva variable conocida como “*momentum*”.
 - *Momentum*: Al incluir este parámetro se introduce un recuerdo capaz de generar cierta inercia en el sistema ya que en $t+1$ se conserva algo del cambio hecho en el paso t . El *momentum* hace que no se produzcan saltos bruscos en los valores de los pesos de



tal forma que el algoritmo descienda suavemente a través de la superficie del error (Santín González & Valiño Castro, 2003).

Al evitar que el peso de una neurona se ajuste en una dirección diferente a la del grupo de neuronas general, se reduce el tiempo computacional, salvando iteraciones inútiles y evitando los mínimos locales.

- **“Sag” y “Slr”**: Basados en el algoritmo Grprop, el cual deriva del rprop básico, definen la tasa de aprendizaje que se cambia de acuerdo a todos los demás. “Sag” se refiere a la menor derivada absoluta, “slr” a la menor tasa de aprendizaje o velocidad con la que aprende la red.

Entre los algoritmos ofertados por el paquete, el más empleado y el estipulado por defecto es el “Backprop”. Pero tras realizar una serie de pruebas se concluye que, frente al problema planteado se obtiene una solución similar independientemente del algoritmo empleado. Por esa razón se decidió emplear en todos los escenarios el “Rprop+”, con el único objetivo de reducir el tiempo computacional. En el caso de los modelos desarrollados con el paquete RSNNS, se ha empleado el mismo algoritmo, pero con unas modificaciones específicas para las redes *Jordan*.

Por último, para el entrenamiento de las redes se ha lanzado el algoritmo de entrenamiento 100 veces. El objetivo de ello es evitar, en la medida de lo posible, mínimos locales. Además, cabe mencionar que en cada lanzamiento la búsqueda se inicia desde un punto aleatorio, por lo que no es probable que se den dos simulaciones iguales.

4.5 Funciones de activación

La función de activación es la característica principal de las neuronas, la que mejor define el comportamiento de la misma. Aunque las funciones de activación no están limitadas a una categoría en particular, algunas de ellas son mayoritariamente empleadas en el entrenamiento y operación de redes neuronales. Desde un punto de vista matemático puede establecerse una categorización de las funciones de activación según sean discontinuas o continuas; al primer grupo pertenecen las funciones escalón y las funciones mixtas, siendo estas últimas una superposición de una función escalón y una función lineal. Su campo de aplicación, por lo menos en la Hidrología y los Sistemas de Recursos Hídricos, se reduce significativamente. La



mayoría de los algoritmos de entrenamiento, de las redes que se emplean en estas disciplinas, están estructurados en términos de las derivadas de las funciones de activación. Con respecto a las funciones continuas, en general, cualquier función puede ser una función de activación adecuada, siempre y cuando sea monótona creciente y esté definida en el intervalo de los posibles valores de las entradas netas (Haykin, 1999)

En una RNA básica se pueden diferenciar tres funciones. La primera es la **función de entrada**, mediante la cual se combinan todos los valores de entrada a una neurona oculta en uno. La función de entrada puede describirse como (Matich, 2001):

$$Input_i = (in_{i1} * w_{i1}) + (in_{i2} * w_{i2}) + \dots (in_{in} * w_{in})$$

Donde:

$Input_i$ = entrada a la neurona oculta i

in_{in} = Dato de entrada de la variable n

w_{in} = Peso sináptico conexión, variable n - neurona i

El operador entre los productos de las entradas brutas y los pesos (en el ejemplo mostrado la sumatoria) puede ser modificado.

La segunda es **la función de activación**. Una neurona biológica puede estar activa (excitada) o inactiva (no excitada), es decir, que tiene un “estado de activación”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado

La función activación calcula el estado de actividad de una neurona, transformando la entrada global en un valor de activación, cuyo rango normalmente va de (0 a 1) o de (-1 a 1). Esto es así, porque una neurona puede estar totalmente inactiva (0 o -1) o activa (1),(Matich, 2001).

Para aprovechar la capacidad de las RNA de aprender relaciones complejas o no lineales entre variables, se recomienda la utilización de funciones no lineales al menos en las neuronas de la capa oculta. En general se utilizará una función sigmoideal (logística o tangente hiperbólica) como función de activación en las



neuronas de la capa oculta. Siendo, dichas funciones, las más empleadas en estudios hidrológicos (Maier & Dandy, 2000) (Camilo & Rivera, 2002).

El último componente que una RNA es **la función de salida**. La elección de la función en las neuronas de la capa de salida dependerá del tipo de tarea impuesta. En tareas de clasificación, las neuronas normalmente toman la función de activación sigmoideal. En cambio, en tareas de predicción o aproximación de una función, generalmente las neuronas toman la función de activación lineal, (Marín Diazaraque, 2007).

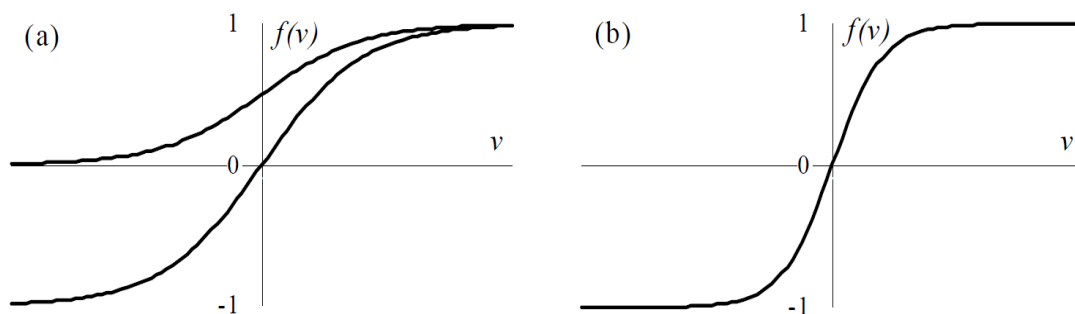
4.5.1 Funciones sigmoideales:

La *función logística* es probablemente la función de activación más usada. Se trata de una función continua que posee un rango comprendido entre [0, 1]. Esto aplicado a las unidades de proceso de una red neuronal artificial significa que, sea cual sea la entrada, la salida estará comprendida [0,1]. La principal limitación de esta función es que no sirve para expresar polaridades, da siempre valores positivos.

$$OUT = \frac{1}{1 + e^{-input_i}}$$

Otra función similar a la logística, la cual da valores de salida comprendidos entre [-1,1] es la *tangente hiperbólica*, dada por la ecuación:

$$OUT = \frac{1 - e^{-input_i}}{1 + e^{-input_i}}$$



Gráfica 2 funciones sigmoideales: (a) logística; (b) tangente hiperbólica



4.6 Estructura de la red

Una vez determinadas las tipologías de red a emplear, es necesario especificar el número de capas y el número de neuronas por capa. La mayoría de autores coinciden en que no existe una forma concreta de saber la estructura exacta que de la mejor respuesta.

Existe un teorema que demuestra que un MLP con un número de neuronas suficiente en la capa oculta puede calcular cualquier función con el grado de aproximación que sea necesario [(Funahashi, 1989);(K. Hornik, Stinchcombe, & White, 1989);(White, 1990)]. Para funciones muy complejas o con discontinuidades, haría falta un MLP con dos capas ocultas (Cybenko, 1988). Tras consultar trabajos similares al presente TFM, se asume que con una sola capa oculta la red será capaz de aproximar la función objetivo con la precisión deseada.

Después de seleccionar el número de capas de la red, se debe elegir el número de nodos ocultos teniendo en cuenta que si la red es muy pequeña no será capaz de formar un buen modelo del problema. Por otro lado, si la red es muy grande corre el peligro de sobrentrenarse, como se comentó en el apartado 3.5. Normalmente se empieza con una red con pocas unidades en la capa oculta y se van añadiendo nodos en función de los resultados observados (Santín González & Valiño Castro, 2003).

Con el objetivo de acelerar y automatizar la elección de la estructura y teniendo en cuenta que en el proyecto se pretende plantear varios escenarios. Se ha decidido programar en el código un bucle que construyese (**n*10**) redes para cada escenario, desde la más sencilla con una variable de entrada y única neurona oculta hasta la más compleja con **n inputs** y diez neuronas ocultas. Además, para cada modelo se lanza el algoritmo de entrenamiento diez veces, de esta forma reducimos las posibilidades de que la respuesta corresponda a un mínimo local. Mediante la ejecución del bucle se entrena y se valida cada modelo. A continuación, se calcula el error cuadrático medio y se grafican los resultados. Por último, se identifica manualmente la estructura que responde mejor al problema planteado.



Ilustración 4 Esquema del funcionamiento del código para la elección de la estructura neuronal. Fuente: Elaboración propia

4.7 Validación de la RNA

Para comprobar el funcionamiento de la red diseñada se utiliza el método de validación cruzada (*cross-validation*) introducido por Cover, (1969). Dicho método consiste en dividir la muestra de datos en dos partes; una parte se utiliza como conjunto de entrenamiento para ajustar los pesos sinápticos de la red. Y la otra parte, llamada conjunto de validación o test, se utiliza para estimar el error de generalización, es decir, el ajuste del modelo durante la fase de ejecución o fase de validación. Es importante que el conjunto de validación no contenga datos utilizados en la fase de entrenamiento.

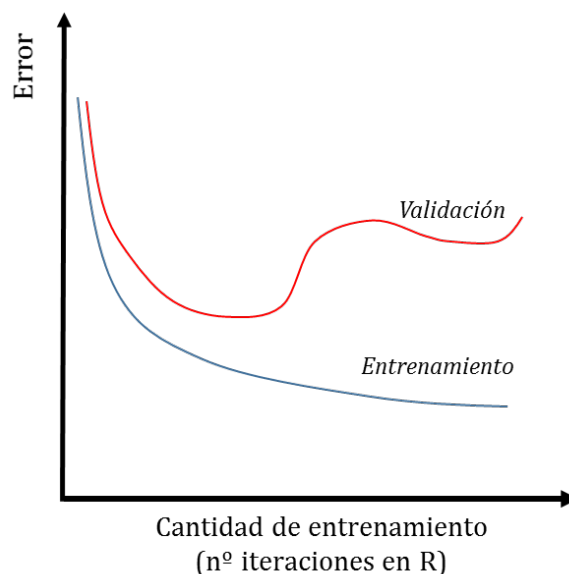
Se ha utilizado aproximadamente el 60% de los datos (1000 patrones aprox.) para entrenar la red y el 40% restante como conjunto de validación (800 patrones aprox.). Aunque en la bibliografía consultada se recomienda emplear el 90% de la



muestra para entrenar la red y el resto para validar, en caso de estudio interesa conocer cómo responde la red durante todo un año.

El algoritmo de entrenamiento ajusta los pesos de la red hasta minimizar el error entre la entrada y la salida. Pero el error sobre el conjunto de validación no decrece ilimitadamente. Normalmente acaba alcanzando un punto a partir del cual crece, lo que indica que a partir del mismo la RNA está sobreentrenada. Por esa razón el proceso de entrenamiento debe finalizar cuando se alcance el primer mínimo de la función del error de validación.

El número de lanzamientos del algoritmo de entrenamiento está directamente relacionado con localizar la estructura neuronal que ofrezca mejores respuestas. Si solo se entrena la red una vez, se corre el riesgo de haber localizado un mínimo local. De ahí que se incluyan 100 lanzamientos del algoritmo de entrenamiento, una vez elegida la estructura neuronal final.



Gráfica 3 Relación entre la evolución del error del entrenamiento y el error de la validación (Marín Diazaraque, 2007)

4.8 Análisis de Resultados:

Con el objetivo de facilitar el análisis de los resultados se ha calculado el r^2 como índice de ajuste para conocer la capacidad de ajuste de la red diseñada. Y para identificar la desviación entre los resultados y los observado, se ha optado por calcular el error cuadrático medio [MSE] y la raíz cuadrada del MSE [RMSE].



Como se puede observar en el apartado del estado del arte, ambos índices son comúnmente empleados para comprobar la empleabilidad de las RNAs en diferentes ámbitos de estudio. A continuación, se muestran las respectivas ecuaciones.

$$r_{xy}^2 = \left[\frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \sqrt{n \sum y_i^2 - (\sum y_i)^2}} \right]^2$$

$$MSE = \frac{\sum_{i=1}^n (y_i - x_i)^2}{n}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - x_i)^2}{n}}$$

Donde:

r^2 = coeficiente de correlación.

RMSE = Raíz del Error Cuadrático Medio

$\sum x_i y_i$ = sumatoria de los productos de ambas variables.

$\sum x$ = sumatoria de los valores de la variable independiente.

$\sum y$ = sumatoria de los valores de la variable dependiente.

$\sum x_i^2$ = sumatoria de los valores al cuadrado de la variable independiente.

$\sum y_i^2$ = sumatoria de los valores al cuadrado de la variable dependiente.

n = tamaño de la muestra en función de parejas

MSE = Error Cuadrático Medio



5. Caso de Estudio

En el presente apartado, se nombran las características ambientales más relevantes del caso de estudio elegido para el desarrollo del presente TFM. Además, se incluye una pequeña descripción del contexto histórico del Parque Natural y su relación con las poblaciones adyacentes.

La Albufera de València ha sido objeto de numerosos proyectos dedicados a regenerar los ecosistemas degradados a causa de la presión antrópica a la que se ha visto sometida. Entre los proyectos, destacan algunas medidas directas como fueron: la regeneración de las islas interiores (“Matas”), la construcción de colectores para evitar el aporte no controlado de vertidos a la laguna o las jornadas de voluntariado. Pero también, encontramos medidas indirectas o administrativas ligadas a la gestión del paraje. Es en este segundo plano donde las herramientas desarrolladas en este TFM pueden servir para facilitar la toma de decisiones u orientar futuras líneas de gestión.

Por el valor cultural, el compromiso administrativo y la riqueza ecológica del ecosistema, se concluye que la Albufera es un buen caso de estudio para aplicar las técnicas de modelación presentadas.

5.1 Descripción Parque Natural de la Albufera

La información expuesta en el presente apartado puede consultarse en el www.albufera.com y en la Tesis Doctoral de Gabriela Onandia Bieco, (2015).

La Albufera de Valencia es un lago somero y oligohalino (salinidad 1-2 ‰), situado en el PNAV (210 km²), cerca de la ciudad de València. Su superficie es de 2320 ha y tiene una profundidad media de 1.2 m. La tasa media de renovación del agua se ha reducido de 10 a 7.4 año⁻¹ entre 1988 y 2004, debido a la menor entrada de caudales al sistema por cambios en la gestión del agua en su cuenca y por los periodos de sequía en años recientes [(Soria, 2006),(Soria & Vicente, 2002)].

Constituye un área geológicamente deprimida y subsidente, que se ha rellenado con aluviones cuaternarios tras producirse el cierre del antiguo golfo por un cordón litoral. Esta barrera arenosa, cordón litoral o restinga tiene su origen en los aportes de sedimentos fluviales procedentes principalmente del Turia y distribuidos desde



el norte por la corriente de deriva. En ella se puede diferenciar un cordón de dunas delanteras móviles más próximas a la playa y sin apenas vegetación y un cordón de dunas fijas que aparece tras el anterior frente dunar. A continuación de las dunas fijas colonizadas por vegetación arbustiva y arbórea, se localiza la laguna (Ferriol-Gabarda, 2013).

El cordón litoral se halla atravesado por las golgas o canales que comunican el lago con el mar y que regulan los niveles de agua en el lago mediante compuertas. La distancia desde la orilla del lago en su margen Este hasta la orilla del mar es de aproximadamente 1 km (Ferriol-Gabarda, 2013).

Antiguamente el agua que la componía era salada pero los aportes de aguas dulces de barrancos, acequias y surgencias denominadas *ullals* redujeron la salinidad hasta el punto actual.

En la Albufera se localiza uno de los paisajes antropizados tradicionales más interesantes del mediterráneo español y una de las áreas de mayor valor ecológico de la península. Actualmente la mayor parte del agua que entra en el lago proviene de las acequias de riego circundantes y de los campos de arroz que desaguan en el lago. También recibe aguas procedentes de dos EDAR (Pinedo y Albufera Sur).

Además de su declaración como Parque Natural por la Comunidad Valenciana, la Albufera ha sido incluida como área de especial protección en el ámbito comunitario e internacional. Constituye desde abril de 1991 una Zona de Especial Protección para las Aves [ZEPA] según la Directiva Aves (79/409/CEE), está incluida desde mayo de 1990 en la Lista de Humedales de Importancia Internacional del Convenio RAMSAR y posee hábitats y especies recogidos en la Directiva Hábitats (92/43/CEE). También forma parte del listado generado en el Protocolo de Ginebra, de 3 de abril de 1982, sobre zonas especialmente protegidas del Mediterráneo y forma parte de *la Red Natura 2000*. Todos estos reconocimientos implican necesariamente un compromiso de conservación y la convierten en zona preferente para la financiación de las distintas medidas comunitarias, estatales y autonómicas.



5.1.1 Localización

PNAV, localizado a unos 15 kilómetros del extremo sur de la ciudad de Valencia, constituye el mayor lago de la Península Ibérica. El conjunto de municipios que comprenden el territorio del Parque se divide en cuatro comarcas: la Ribera Alta, la Ribera Baja, la Huerta Sur y la Ciudad de Valencia.



Ilustración 5 Imagen satélite de la Albufera de Valencia. Fuente: Image Landsat/ Google Earth

La cuenca hidrográfica de la Albufera se sitúa entre los ríos Turia al norte y Júcar al sur y el drenaje de la cuenca al lago, se realiza fundamentalmente a través de seis barrancos principales: Picassent, Hondo, Berenguera, del Agua, de Montes y, sobre todo, el barranco de Torrent también llamado del Poyo, de Xiva o de Massanassa (Ferriol-Gabarda, 2013).



5.1.2 Eutrofización de la laguna

El rápido proceso de industrialización junto al incremento poblacional de la zona durante los años 1960 resultó en un aumento de entradas de nutrientes al lago y su eutrofización. Este proceso, unido al cambio en las prácticas agrícolas y el consiguiente aumento en los niveles de pesticidas, favoreció la desaparición de los macrófitos sumergidos. Lo cual, consolidó la transición al estado turbio del lago a principios de los años 1970, [(Blanco, 1974);(Benet Granell, 1983);(Vicente & Miracle, 1992)]. Durante esta década, el fitoplancton se estableció como productor primario dominante en el lago en detrimento de los macrófitos [(Blanco, 1974);(Dafauce Ruiz, 1975)]. Seguidamente, la densidad de fitoplancton y especialmente de cianobacterias filamentosas aumentó de manera considerable y durante los años 1980, se llegaron a registrar concentraciones de clorofila-*a* superiores a $800\mu\text{g L}^{-1}$ en el lago [(Vicente & Miracle, 1992);(Miracle M.R., García, & Vicente, 1984);(Romo & Miracle, 1994)].

A principios de los años 1990, se implementó un plan de depuración de las aguas residuales vertidas al lago con el objetivo de mejorar la calidad del agua del mismo. En la misma línea de actuación, se han construido recientemente tres filtros verdes (*Tanecat de la Pipa*, *Tanecat de Mília* y *Estany de la Plana*) en las inmediaciones del lago. A pesar de las medidas mencionadas, en la actualidad el lago permanece en estado turbio y se considera hipertrófico (Usaquén Perilla, Gómez, Gómez, Díaz, & Cortezón, 2012);(Martín, Oliver, Hernández-Crespo, Gargallo, & Regidor, 2013)).

La eutrofización de un ecosistema acuático es el resultado indirecto de desajustes físico-químicos en el sistema. A continuación, en la ilustración 6, se muestra un esquema donde aparecen gran parte de los factores y procesos que desencadenan y mantienen las respuestas del fenómeno de eutrofización.

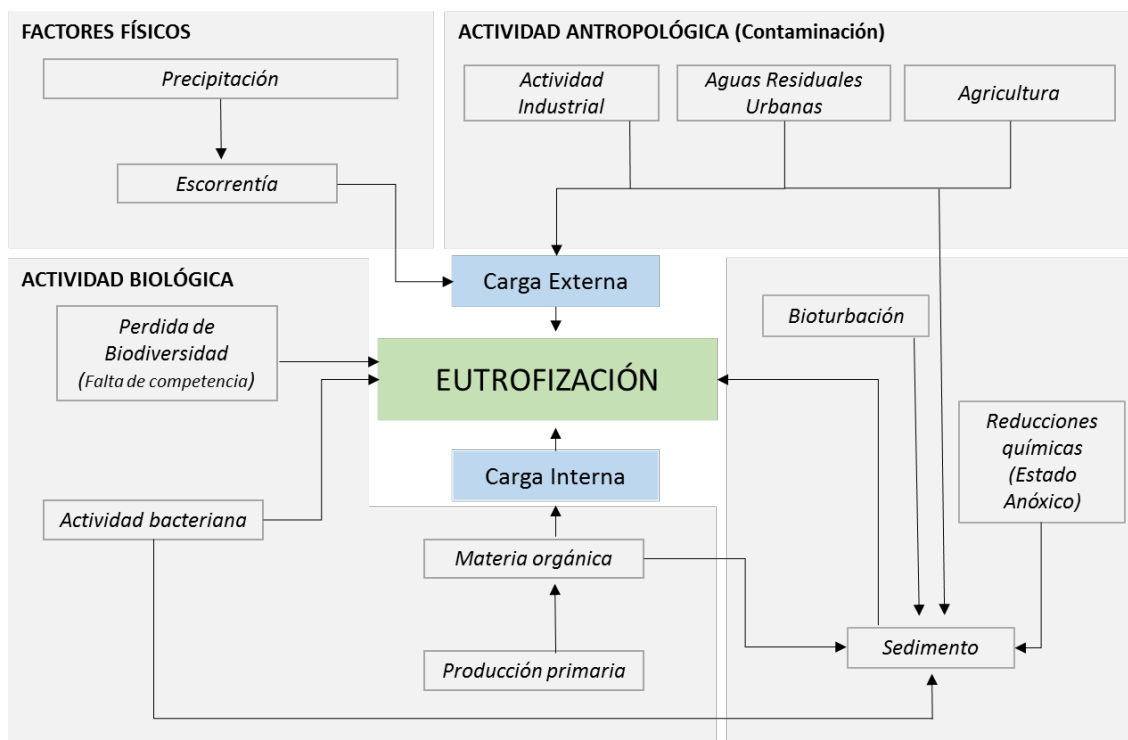
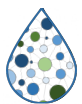


Ilustración 6 Principales factores y procesos que desencadenan y mantienen las respuestas del proceso de eutrofización (Mazzeo et al., 2002)

Son muchos los impactos asociados a la eutrofización de un ecosistema acuático. A continuación, se describen algunos de los más relevantes:

- **Pérdida de biodiversidad:** el crecimiento anormal de las poblaciones de microalgas tiene como consecuencia un primer enturbiamiento del agua. Este fenómeno impide que la luz alcance las zonas más profundas del perfil del lago. La falta de luz provoca la desaparición de la población de macrófitas instaladas en el fondo y con ellas todas las especies asociadas o dependientes. Además, todas las comunidades de peces depredadoras que empleen la visión para detectar las presas se ven afectadas, obligando su desplazamiento hacia otros hábitats. Todo ello produce un empobrecimiento del ecosistema masivo y facilita la intrusión de especies exóticas.
- **Muerte de los individuos por hipoxia:** la hipoxia puede que sea el síntoma más grave asociado a la eutrofización. Este fenómeno se da cuando el *Bloom* de algas y otros microorganismos asociados mueren sedimentando en el fondo de la laguna. El material detrítico acumulado comienza a ser descompuesto por comunidades bacterianas usando el oxígeno disuelto disponible. La



reducción aguda de la concentración de oxígeno en el agua suele ir asociada a la muerte masiva de peces y especies bentónicas.

- *Problemas estructurales:* por colmatación y obstrucción de los canales de riego por las malas hierbas acuáticas (Jacinto de agua en Andalucía).
- *Limitación del recurso:* inhabilitación del uso del agua por degradación de la calidad del recurso. Como puede ser la aparición de malos olores, debido a la descomposición de las algas o la liberación de toxinas por determinadas especies.

La caracterización del estado trófico de un ecosistema lacustre como la Albufera, se puede realizar de forma sencilla visualmente. Es posible observar ciertas características generales que permiten diferenciar rápidamente una situación oligotrófica de un estado eutrófico. El caso de estudio cumple con todos los rasgos de la segunda situación.

Tabla 4 Caracterización del estado trófico de un ecosistema lacustre

PARÁMETRO	MASA DE AGUA TIPO LAGO	
	OLIGOTRÓFICO	EUTRÓFICO
Características Biológicas		
<i>Producción de plantas y animales</i>	Baja	Alta
<i>Número de especies</i>	Numerosas	Pocas
<i>Niveles generales de biomasa</i>	Bajos	Altos
<i>Aparición de proliferación de algas</i>	Rara	Frecuente
<i>Cantidad relativa de algas verdes y verdes-azules</i>	Baja	Alta
<i>Grado distribución de algas</i>	En el hipolimnion, en masas de agua estratificadas térmicamente	Por lo general sólo en las aguas superficiales
<i>Crecimiento de plantas acuáticas en el litoral del lago</i>	Puede ser escaso o abundante	Aparición de algas filamentosas y desaparición de micrófitos
<i>Emigración diaria de algas</i>	Grande	Limitada



Tipos característicos de peces	Peces que viven en las profundidades y agua fría	Peces que viven en la superficie y agua caliente
Características Físico-Químicas		
Turbidez	Baja	Muy Alta
Color y Olor	Bajo	Alto

Otro método para clasificar de forma rápida y sencilla el estado trófico de un lago es mediante el cálculo del Índice de Estado Trófico [TSI] publicado por Carlson,(1977). Dicho índice propone tres opciones para su cálculo: la concentración de fósforo total [PT], la concentración de la clorofila-*a* [Chl-*a*] o a partir de la medición de turbidez con el disco Secchi [SD].

$$TSI (chl - a) = 9.81 \ln(Chl - a) + 30.6$$

$$TSI (PT) = 14.42 \ln(PT) + 4.15$$

$$TSI (SD) = 10 * \left[\frac{\ln(SD)}{\ln(2)} \right]$$

Según el resultado de esta ecuación el lago obtenemos las siguientes clasificaciones:

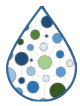
Tabla 5 Clasificación del estado trófico del ecosistema en función del TSI (Carlson, 1977)

Rango	ESTADO TRÓFICO
TSI<25	Ultraoligotrófico
25<TSI<35	Oligotrófico
35<TSI<55	Mesotrófico
55<TSI<70	Eutrófico
70<TSI	Hipereutrófico

5.2 Modelación clorofila-*a*

Los organismos fotosintéticos emplean los pigmentos para absorber la luz y poder llevar a cabo las reacciones fotosintéticas. La clorofila-*a* capta todas las longitudes de onda de la luz visible excepto el verde, el cual es reflejado.

Aunque existen especies que emplean otros pigmentos como los carotenoides para sintetizar las sustancias esenciales para su metabolismo, la mayoría de las microalgas emplean la clorofila-*a* para desarrollarse. Dicha dominancia es la razón



por la cual la concentración de clorofila-*a* es uno de los parámetros más importantes en el estudio de las características y el estado del agua(Desortová, 2007). Se puede relacionar de forma significativa la concentración de clorofila-*a* con el nivel de eutrofización de un ecosistema acuático.

Las microalgas son susceptibles a sufrir incrementos agudos en sus poblaciones cuando se producen desajustes en el sistema, como puede ser un aporte incontrolado de nutrientes. La modelación de la clorofila-*a* de un ecosistema acuático aporta información sobre si se encuentra o no en equilibrio ecológico y permite conocer el estado trófico del mismo.

En la ilustración 7, se muestra el esquema conceptual del modelo mecanicista Humecal desarrollado por Edgar Belda, (2015). En él se puede identificar la relación directa o indirecta de cada una de las variables con la concentración de clorofila-*a* del sistema.

MODELO HUMECAI

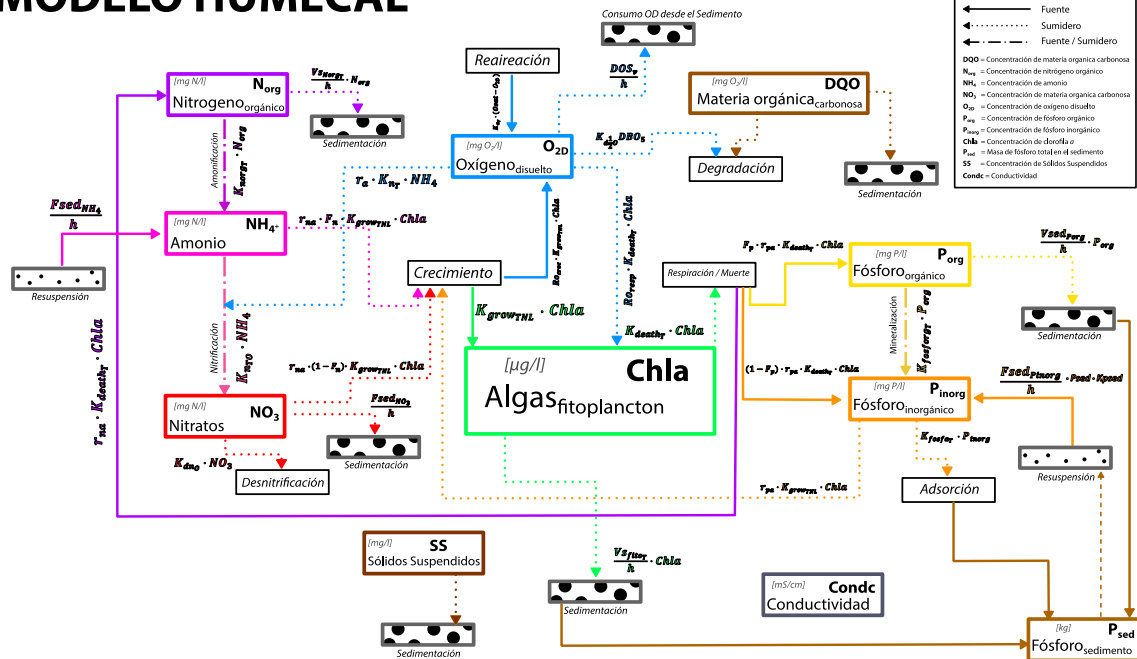


Ilustración 7 Esquema del modelo Humecal (Belda, 2015)

La complejidad de los sistemas modelados obliga a los expertos a simplificar las reacciones físicas, químicas y biológicas entre los parámetros más relevantes, (Belda, 2015).



En la naturaleza encontramos nutrientes que actúan como limitantes del crecimiento de las poblaciones de microalgas. Aunque hay diferencias entre especies, normalmente la concentración de fósforo inorgánico es la que condiciona el desarrollo de las comunidades fotosintéticas acuáticas. Para representar estos comportamientos se debe calibrar el modelo e identificar los parámetros que afectan más la variable objetivo. La calibración de las RNAs se realiza mediante el entrenamiento de las mismas y a partir de la organización de los pesos sinápticos se puede identificar que *inputs* influyen más en el comportamiento *output*.

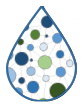
5.3 Escenarios propuestos

En el presente TFM se han planteado cuatro escenarios, todos ellos se aplican sobre el mismo caso de estudio. La única diferencia es el desfase temporal entre los datos empleados como *inputs* y el valor que se desea estimar. Para cada uno de ellos se ha desarrollado una RNA específica.

Inicialmente la resolución de los escenarios se plantea mediante la construcción de redes del tipo MLP. En el caso de que los resultados no sean satisfactorios se ha propuesto desarrollar el modelo mediante redes recurrentes simples o también conocidas como red *Jordan*.

En el *primer escenario*, no existe desfase temporal. Es decir, tanto los *inputs* como las concentraciones de clorofila-*a* estimadas. En el *segundo escenario*, se aplica un desfase de siete días entre los valores empleados como *inputs* y el *output*. En el *tercer escenario*, la estimación se realiza con quince días de antelación. Y en el *cuarto escenario*, el desfase es de 30 días.

El objetivo de ir adelantando temporalmente la estimación de la concentración de clorofila-*a* en la laguna, es conocer si la red es capaz de ajustarse a la dinámica del sistema analizado. Es decir, se pretende evaluar la capacidad de las RNAs para adaptarse y representar la relación entre la situación del sistema estudiado en el instante t y las concentraciones de clorofila-*a* en $(t+7)$, $(t+15)$ y $(t+30)$.



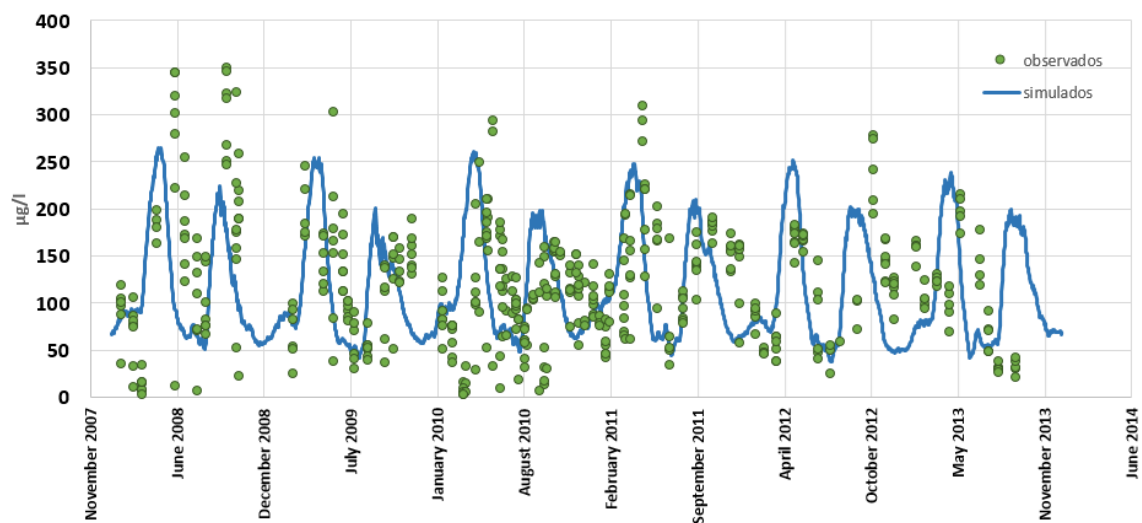
5.4 Muestra de datos

Para la implementación de RNAs se precisa amplias muestras de datos observados. Al no disponer de dicho recurso, se ha optado por recurrir a los resultados de un modelo mecanicista.

Los modelos mecanicistas de mezcla completa ya suponen una simplificación de la realidad. Simplificación necesaria para poder representar las reacciones químicas, físicas y biológicas que se dan en los sistemas modelados, (Belda, 2015). Asumiendo que no se tratan de datos observados, el máximo rendimiento que se puede obtener de la RNA será igual al del modelo mecanicista.

Se ha empleado los resultados del modelo Humecal para el entrenamiento y la validación de las RNAs. Humecal es una herramienta para el cálculo de la calidad del agua, enfocada a determinar concentraciones en sistemas eutróficos (Belda, 2015). El buen comportamiento y los buenos ajustes globales obtenidos tras su implementación en el *tancat* de Milia y en la Albufera, han motivado que se incluyan nuevas simulaciones en el Plan Especial de l'Albufera.

La disponibilidad de nuevas series de datos observados y el interés por parte de los órganos competentes, han motivado que durante los primeros meses del 2017 se recalibre el modelo Humecal. En la gráfica 4 se pueden observar los resultados del modelo y los datos observados de clorofila-*a* en la laguna.

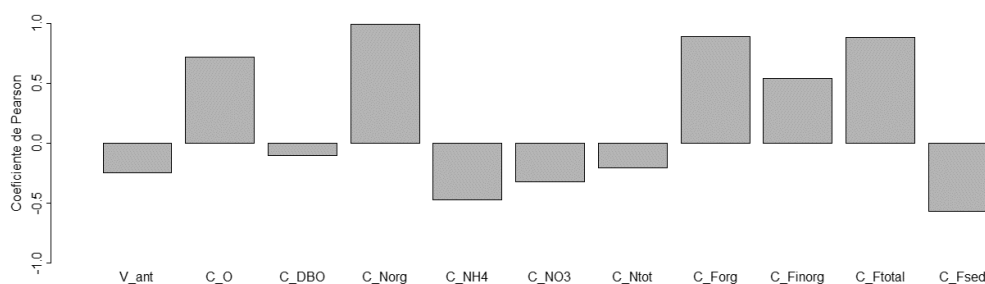


Gráfica 4 Resultados de la calibración de Humecal en la Albufera-2017



Cabe destacar que, la mayoría de los modelos mecanicista estiman el incremento de la variable objetivo respecto a la situación anterior. Pero en el caso de las RNAs, a no ser que sean redes recurrentes, no existe ninguna conexión entre el patrón anterior y el posterior. Es decir, la estimación de cada valor es independiente al resto. Durante el desarrollo de los modelos presentados en este TFM, no se emplea términos de tiempo porque las redes trabajan con patrones.

Para la **elección de los inputs** de cada escenario se realiza un primer análisis para identificar las variables que tienen relación directa o indirecta con la dinámica poblacional de las microalgas y pueden aportar mayor información a la RNA. La evaluación de los posibles *inputs* se ha realizado mediante el cálculo de los coeficientes de Pearson de cada parámetro respecto a la concentración de clorofila-*a*.



Gráfica 5 Coeficiente de Pearson de los posibles inputs

Tabla 6 Leyenda grafica 5

posibles Inputs	
V_ant	<i>Volumen de la laguna en (t-1)</i>
C_O	<i>Carga de oxígeno</i>
C_DBO	<i>Carga de Demanda Biológica de Oxígeno</i>
C_Norg	<i>Carga de Nitrógeno orgánico</i>
C_NH4	<i>Amonio</i>
C_NO3	<i>Carga de Nitrato</i>
C_Ntot	<i>Carga de Nitrógeno total</i>
C_Forg	<i>Carga de Fósforo Total</i>
C_Finorg	<i>Carga de Fósforo Inorgánico</i>
C_Ftotal	<i>Carga de Fósforo Total</i>
C_Fsed	<i>Carga de Fósforo en el Sedimento</i>



Las variables incluidas en los archivos de lectura del código fueron aquellas que no se encontraban en el rango $[-0.3, 0.3]$ del coeficiente de Pearson, es decir que el r^2 sea superior a 0.1 aproximadamente. Una vez identificados los posibles *inputs*, se ejecuta el código y se obtiene la combinación de *inputs* que tiene mejor respuesta.

Seguidamente se realiza una descripción de cada uno de los parámetros incluidos como *inputs* en alguna de las redes construidas, especificando los fuente-sumidero de los que dependen.

- *Ciclo del nitrógeno:*

El nitrógeno es uno de los elementos más importantes en química orgánica, debido a que lo encontramos en moléculas como las proteínas y ácidos nucleicos. Pero en los hábitats acuáticos cobra especial relevancia por ser indispensable para el correcto desarrollo de los organismos fotosintéticos. En condiciones naturales, los compuestos nitrogenados del agua provienen fundamentalmente de la descomposición de materia orgánica. Pero la presión antrópica ha aumentado las entradas de nitrógeno a los sistemas acuáticos. Los vertidos urbanos, el incremento en el uso de fertilizantes sintéticos o ciertas actividades industriales, han producido un desequilibrio en el ciclo de este elemento.

Los compuestos nitrogenados incluidos como *inputs* en las redes desarrolladas son:

- **Nitrógeno orgánico:** Forma parte de la materia orgánica y su principal fuente es la muerte de todo tipo de organismos acuáticos y alócatenos. Cuando las bacterias empiezan a descomponer los compuestos orgánicos complejos se produce la amonificación, proceso en el cual se transforma el nitrógeno orgánico en amonio.
- **Amonio:** Es una de las dos formas de absorción de nitrógeno para los organismos fotosintéticos. El crecimiento de las poblaciones de algas se ve afectadas directamente por cambios en las concentraciones de este compuesto. La fuente natural más importante de amonio en los ecosistemas acuáticos es la descomposición de nitrógeno orgánico. En la actualidad la incorporación de fertilizantes industriales en la agricultura y el mal uso de estos, incorpora una fuente de amonio extra a los



ecosistemas. Los sumideros más relevantes son el proceso de nitrificación y el consumo por parte de los organismos fotosintéticos

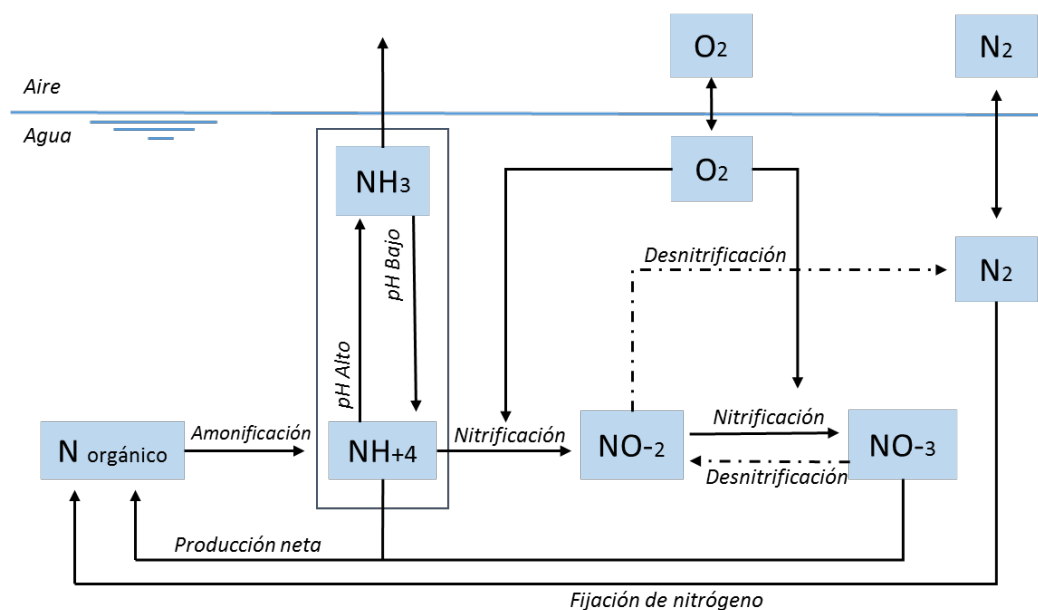


Ilustración 8 Ciclo del nitrógeno en el agua. (Las líneas discontinuas indican los procesos que se desarrollan en condiciones anaeróbicas), (Chapra, 1997).

- **Ciclo del fósforo:**

Al igual que el nitrógeno el fósforo es un componente esencial para el desarrollo de los organismos. Su reserva fundamental en la naturaleza es la corteza terrestre. Mediante los procesos de meteorización y el posterior arrastre por parte de las escorrentías superficiales, el fósforo entra en los ecosistemas acuáticos de forma natural. Generalmente, es el principal limitante para la producción primaria de los sistemas.

El aumento descontrolado en la carga de entrada a causa de las actividades antrópicas produce un desajuste y puede provocar un crecimiento agudo de las poblaciones de microalgas.

El entrenamiento de las redes neuronales se ha realizado aportando datos de los siguientes compuestos fosfatados:

- **Fósforo total:** este parámetro es el producto de la suma de todos los compuestos inorgánicos y orgánicos que contienen fósforo. En el modelo Humecal se incluye también el contenido en fósforo de la comunidad de microalgas.



- **Fósforo orgánico:** Compuestos orgánicos fosfatados en la columna de agua. La principal fuente de fósforo orgánico particulado es la muerte de los organismos acuáticos y alóctonos. El mayor sumidero es la descomposición por microorganismos en compuestos orgánicos disueltos y posteriormente en inorgánicos más simples.
- **Fósforo inorgánico:** Tan solo una fracción pequeña de este fósforo se encuentra en su forma soluble, iones ortofosfatos bivalentes y trivalentes, que pueden asimilar los organismos fotosintéticos. En un sistema no alterado, las principales entradas de fósforo inorgánico provienen de la descomposición de la materia orgánica y por arrastre de la escorrentía superficial. Su limitada disponibilidad impide la proliferación de microalgas y la consecuente eutrofización de los sistemas. Pero en la actualidad, la aparición de fertilizantes fosfatados industriales y las malas prácticas agrícolas han incrementado los niveles de fósforo en los sistemas. Las únicas formas de sumidero de fósforo en los ecosistemas acuáticos son, la fijación temporal en la biomasa algal y la precipitación química en el sedimento.
- *Concentración de oxígeno:*

En los sistemas poco productivos la concentración de oxígeno disuelto está determinada principalmente por factores físicos como la temperatura y su influencia en la solubilidad de los gases. No se observan diferencias muy notorias entre las capas superficiales y profundas (Mazzeo et al., 2002).

El aumento de la productividad promueve diferencias verticales en la concentración de oxígeno, registrándose valores elevados en superficie producto de la actividad fotosintética del fitoplancton y valores cercanos a cero en el fondo como resultado de la degradación de la materia orgánica (Wetzel, 2001). El balance entre fotosíntesis y respiración en periodos de intensa actividad fotosintética, genera cambios diarios en la concentración de oxígeno, con valores de sobresaturación durante el día y déficit durante la noche. El aumento de la productividad primaria también determina variaciones diarias de pH en la columna de agua (Mazzeo et al., 2002).



5.5 Resultados

A continuación, se muestran los resultados obtenidos en cada uno de los escenarios. Especificando detalles de los modelos generados como la estructura, los pesos sinápticos de las neuronas tras la fase de entrenamiento, la raíz cuadrada del error cuadrático medio [RMSE] y el r^2 durante la ejecución de los modelos.

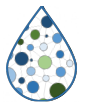
5.5.1 Escenario_01

La característica definitoria del presente escenario es que la estimación de la variable objetivo se realiza en el mismo instante t que las variables *inputs*. En este caso la elección de las variables de entrada o *inputs*, se ha realizado en primer lugar mediante la ejecución del código programado. En el código las variables se van añadiendo una a una a la red, se entrena cada uno de los modelos y se comparan los resultados. Tras evaluar los resultados se ha comprobado que el MSE de los modelos se reduce drásticamente al añadir la variable fósforo inorgánico como *input*.

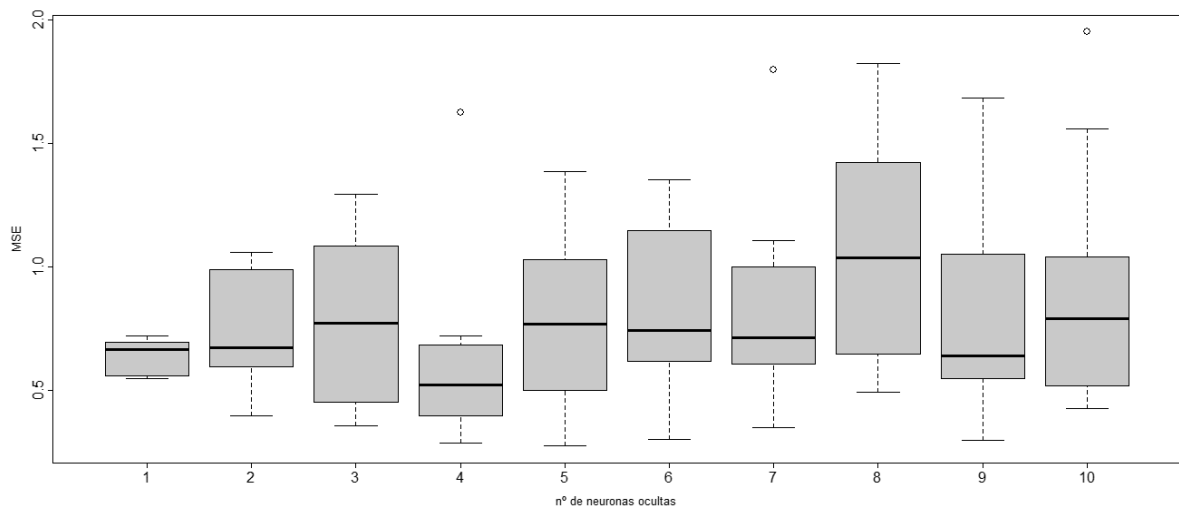
El orden de inclusión de los *inputs* está determinado por el índice de correlación de Pearson calculado para cada variable respecto al *output* del modelo (clorofila- a). Situándose en primer lugar la variable con mayor correlación y descendiendo el índice a medida que se incluyen variables. El fósforo inorgánico se localiza en la sexta posición. Por delante de él se encuentran variables como la concentración de fósforo total o la carga de fósforo biodisponible en el sedimento.

Para simplificar en la medida de lo posible la red, se ha comprobado que si se eliminan algunos de los *inputs* incluidos antes de la entrada de la variable fósforo inorgánico se mantiene los valores de MSE. Los *inputs* eliminados son:

- Concentración de fósforo total, en el modelo Humeccal esta variable se incluye la concentración de fósforo orgánico, la concentración de fósforo inorgánico y el contenido de fósforo de las comunidades de microalgas. Desglosando la variable, su componente orgánico e inorgánico ya están incluidos en la red por lo que no aporta mayor información.
- Carga de fósforo total biodisponible en el sedimento, esta variable es muy difícil de medir en campo por lo que durante la fase de ejecución de la red dificultaría su uso.



En el siguiente gráfico se pueden observar la dispersión, el mínimo, el máximo y los tres cuartiles de los resultados obtenidos tras lanzar cada modelo diez veces. En este caso los *inputs* son fijos y solo se modifican el número de neuronas ocultas del modelo. El mínimo error se obtiene con cuatro neuronas ocultas. Se puede observar una tendencia ascendente respecto al error del modelo cuando se va aumentando el tamaño de la red. Esto puede estar relacionado una sobreparametrización del modelo, fenómeno que suele ir acompañado de un sobreentrenamiento de la red.



Gráfica 6 Rango del MSE de diez redes (variando el número de neuronas ocultas)

Tras la fase de entrenamiento y la optimización de los pesos sinápticos, la estructura funcional de la RNA para el escenario 1, está compuesta por cuatro *inputs*, cuatro neuronas ocultas y un *output* (Ilustración 9).

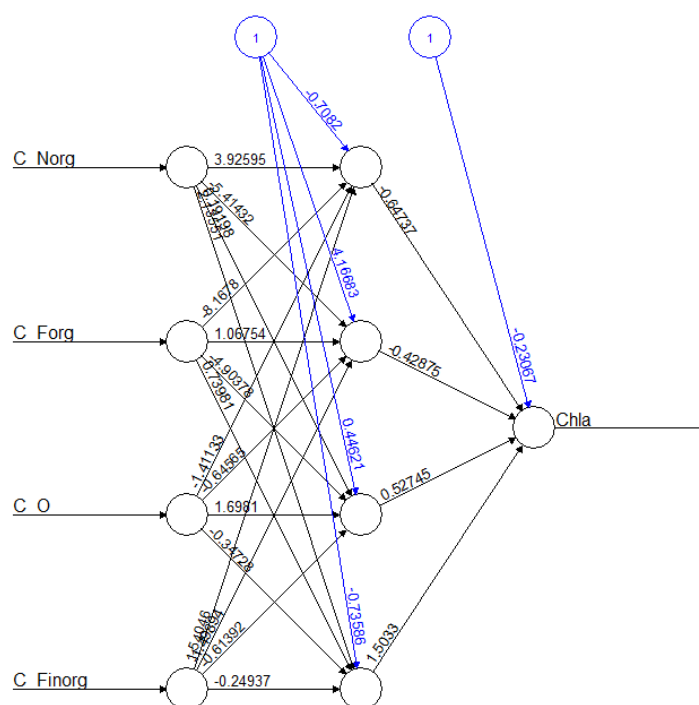


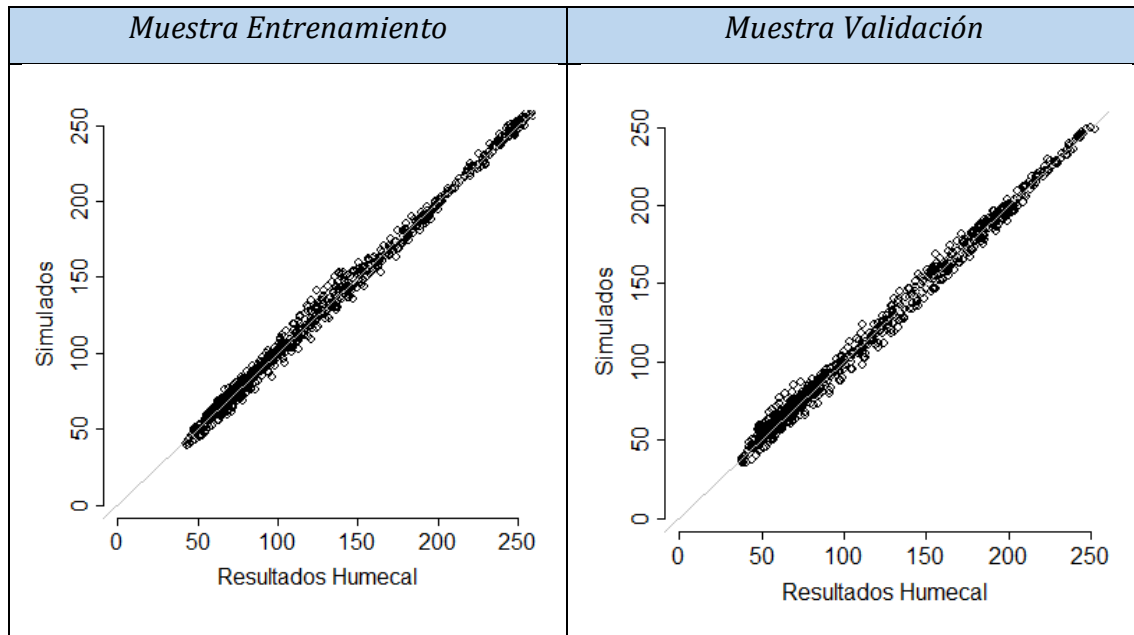
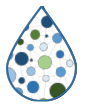
Ilustración 9 Red Neuronal Artificial- Escenario_01

A continuación, se muestran los pesos sinápticos de la red una vez alcanzada la optimización, tras lanzar el algoritmo de entrenamiento 100 veces.

Tabla 7 Pesos de la RNA-Escenario_01

		Neuronas ocultas				
INPUTS		1º	2º	3º	4º	Bias
Pesos, 1º CAPA	C_Norg	3.92	-5.41	0.19	2.13	-0.70
	C_Forg	-8.17	1.06	-4.90	0.73	4.17
	C_O	-1.41	-0.64	1.69	-0.35	0.45
	C_Finorg	1.54	-1.49	-0.61	-0.24	-0.74
Pesos, 2º CAPA		-0.65	-0.43	0.53	0.50	-0.23

En la gráfica 8 se presenta el ajuste obtenido en forma de resultados de concentración de clorofila-*a* del modelo Humecal vs. Concentración clorofila-*a* simulada con la RNA. Cuanto más se aproximen los resultados a la línea de 45°, mejor será la simulación. Se observa como los resultados son satisfactorios. El r^2 de la simulación realizada con la muestra de validación es de 0.994. Por lo que se considera que la red está bien parametrizada y entrenada.



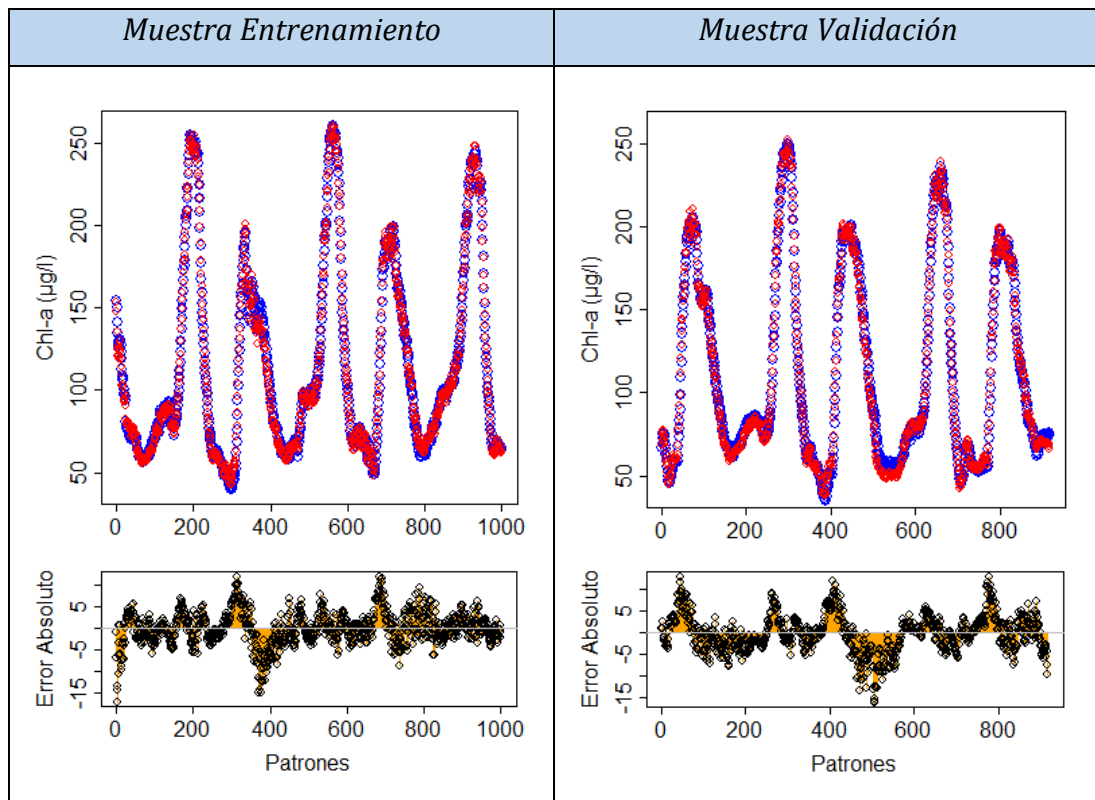
Gráfica 7 Ajuste de la RNA-Escenario_01

En el gráfico 9 se puede observar la estimación de la RNA y los resultados del modelo Humecal, para cada uno de los conjuntos de datos empleados. Los datos simulados con la red se ajustan bien a los obtenidos por Humecal. La RNA precisa de un número menor de parámetros para responder de forma similar al modelo mecanicista. Por lo que se ha simplificado el modelo satisfactoriamente.

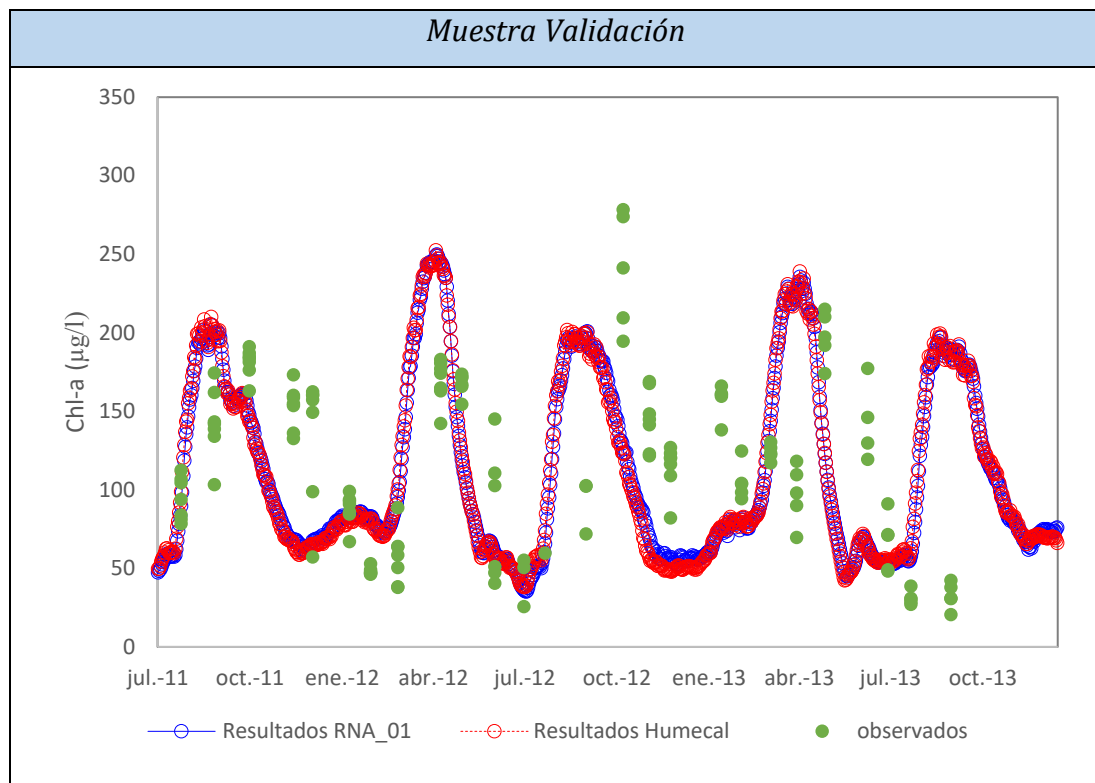
El RMSE del modelo durante la fase de validación o ejecución es de 4.37 μg de Chl-*a*/l. La pérdida de calidad entre los resultados de la muestra de entrenamiento y la muestra de validación son mínimos, ya que el error absoluto en ambos casos se distribuye en el mismo rango, de ± 15 μg Chl-*a*/l.



- Azul: Resultados RNA-Escenario_01
- Rojo: Resultados Humecal



Gráfica 8 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_01



Gráfica 9 Representación datos observados, resultados Humecal y resultados RNA_01



Una vez ejecutada la red se ha representado en la gráfica 9 los datos de clorofila-*a* observados en la laguna durante el periodo 2011-13, junto con los resultados del modelo Humecal y la estimación de la RNA_01.

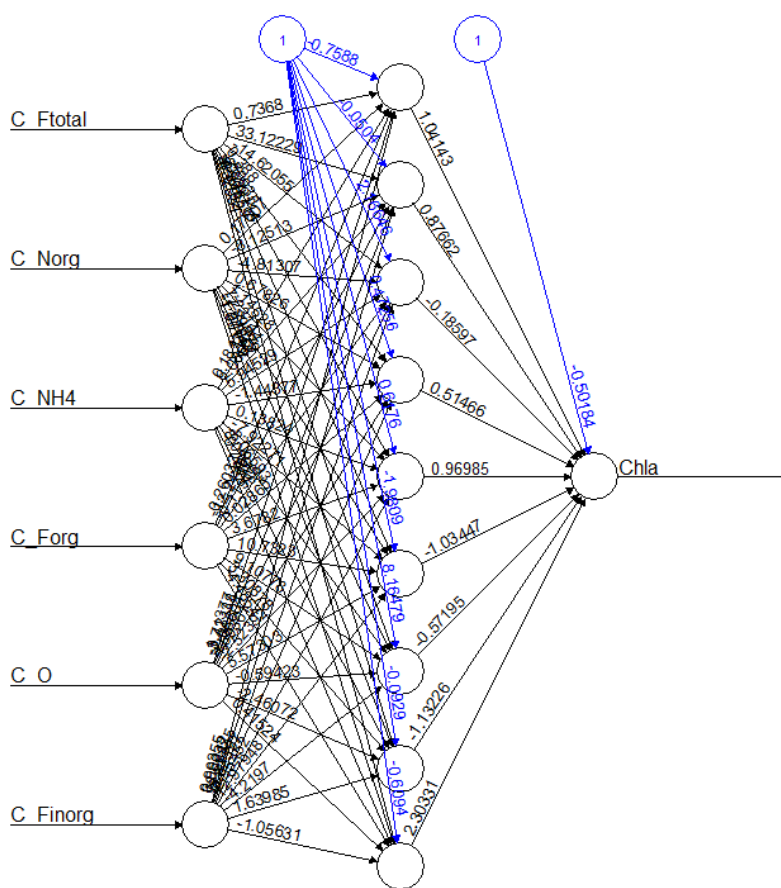
Teniendo en cuenta la calidad de los resultados del modelo, se ha considerado innecesaria la aplicación en este escenario de una red del tipo recurrente.

5.5.2 Escenario_02

Las iniciativas para evitar, reducir y controlar los *blooms* de microalgas no pueden aplicarse una vez iniciada la proliferación de los microorganismos. Es necesario anticiparse a este tipo de eventos para minimizar los daños ambientales, económicos y culturales.

Los valores máximos de clorofila-*a* en la Albufera se dan en los meses de abril y septiembre. Mediante el entrenamiento de la segunda red, se pretende estimar la concentración de la variable objetivo con siete días de antelación. Para ello se ha forzado un desfase del periodo mencionado entre los datos de entrada y los de salida.

Una vez preparados manualmente el archivo de entrada en formato “.csv”, se han cargado los datos en el código y se ha ejecutado para seleccionar la estructura que ofrece un mejor ajuste. A diferencia con la red del Escenario_01, en este caso el modelo precisa de mucha información para poder representar la relación entre las concentraciones de los inputs en t y la concentración de clorofila-*a* en $t+7$. Tras evaluar los resultados y comparar los modelos generados, se concluye que la red con seis inputs y nueve neuronas ocultas es la que da mejor respuesta.

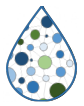


Gráfica 10 Red Neuronal Artificial- Escenario_02

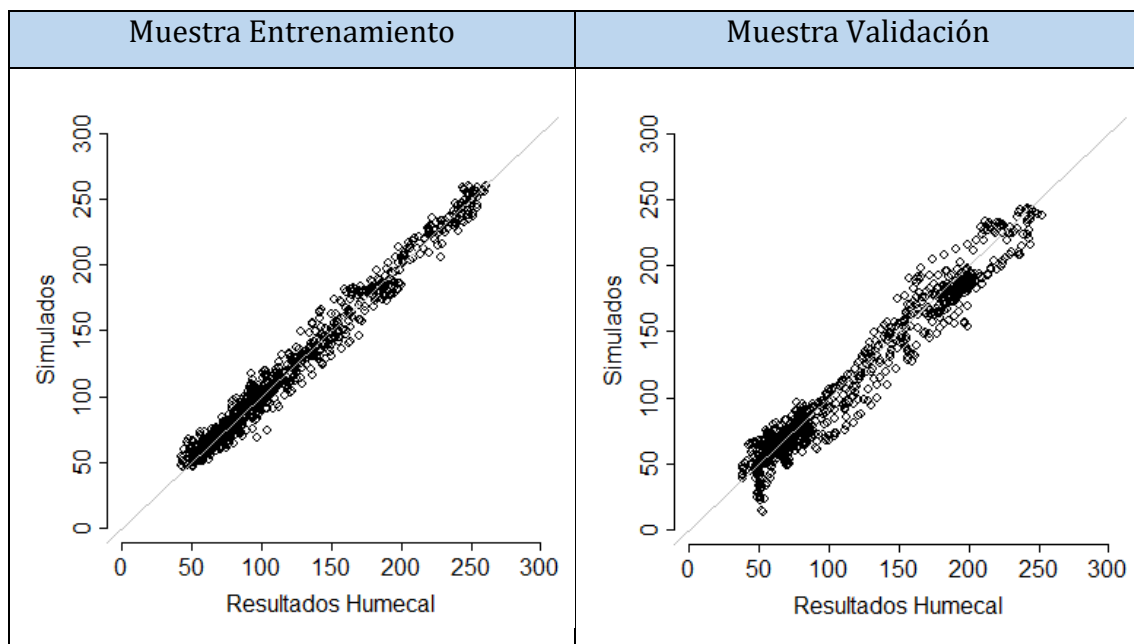
A continuación, se muestran los pesos sinápticos de la red una vez alcanzada la optimización. Para evitar mínimos locales, se ha lanzado el algoritmo de entrenamiento 100 veces.

Tabla 8 Pesos de la RNA-Escenario_02

		Neuronas ocultas								
		1º	2º	3º	4º	5º	6º	7º	8º	9º
Pesos, 1º CAPA	C_Ftotal	0.74	33.12	-14.62	0.39	1.60	-22.89	-4.94	-8.62	4.33
	C_Norg	0.18	-8.13	-4.81	0.68	1.15	7.23	-7.29	-1.28	-1.00
	C_NH4	0.18	6.09	5.95	-1.44	0.14	-7.92	-8.07	8.02	-0.80
	C_Forg	-0.26	-12.24	-0.12	0.03	3.68	10.73	9.11	3.31	-4.49
	C_O	-1.71	-3.42	-2.39	4.89	-2.52	5.57	-0.59	-2.46	0.42
	C_Finorg	0.90	3.57	18.27	-8.62	-2.95	12.98	-4.22	1.64	-1.06
	Bias 1º	-0.76	-0.05	2.17	0.47	0.62	-1.98	8.16	-0.09	-0.61
Pesos, 2º CAPA		1.04	0.88	-0.19	0.51	0.97	-1.03	-0.57	-1.13	2.30
Bias 2º		-0.52								



Al igual que en el Escenario 1, el orden de inclusión de los *inputs* está determinado por el valor de su índice de correlación de Pearson. El cual aporta información de la correlación lineal entre dos variables. Las variables de entrada a la red son: fósforo total, fósforo orgánico, fósforo inorgánico, nitrógeno orgánico, amonio y el oxígeno disuelto. Se han descartado otras variables como la carga de fósforo en el sedimento por la dificultad en su estimación. Todos los parámetros empleados, se puede analizar *in situ* o *ex situ* de forma relativamente sencilla.



Gráfica 11 Ajuste de la RNA-Escenario_02

En este caso una red con pocas neuronas ocultas y pocos *inputs*, como la del Escenario_01, no es lo suficiente flexible para representar el problema planteado. Aunque se ha provisto de mayor número de parámetros, si se compara con el caso anterior el modelo pierde ajuste. Aun así, la nube de puntos se distribuye de forma homogénea en torno a la línea de 45°, la cual corresponde con un ajuste perfecto.

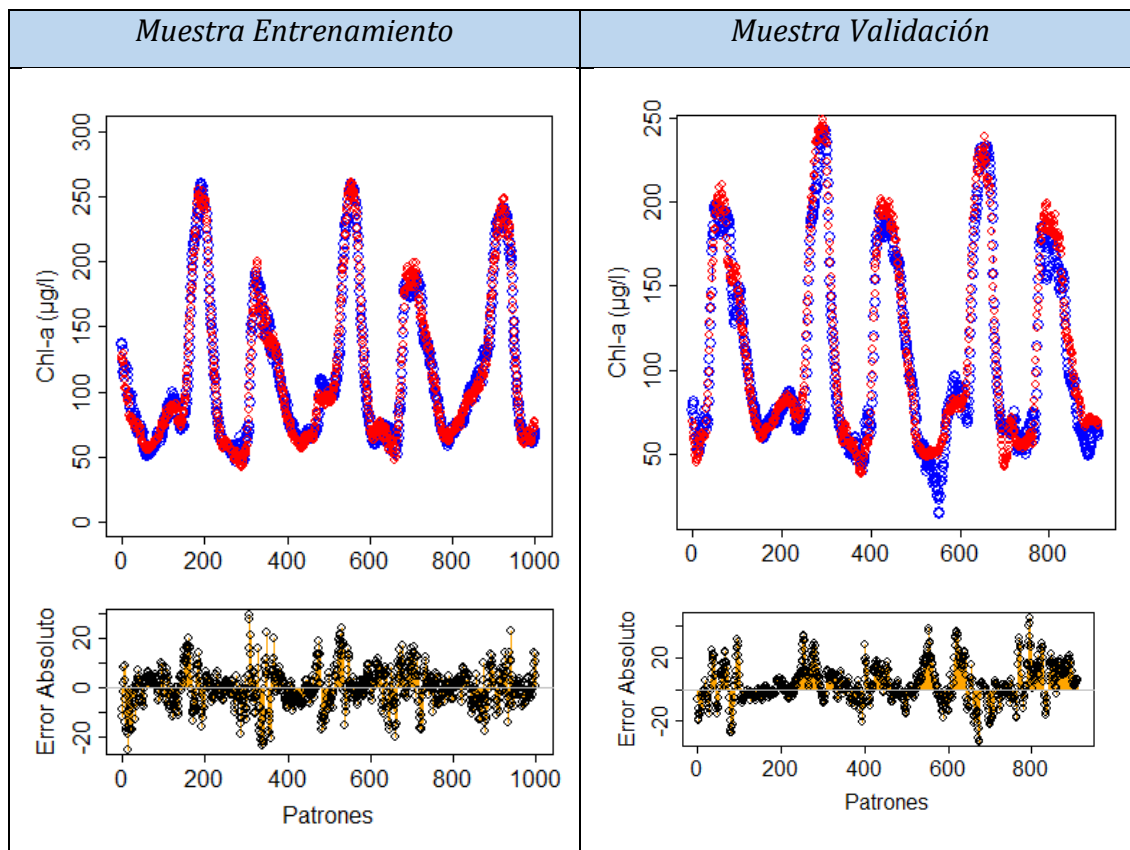
La simulación realizada con la muestra validación, tiene un r^2 de 0.957. Los resultados de la simulación siguen siendo satisfactorios, RMSE menores de 13 μg de Chl-a/l.

Respecto al rango en el que se distribuye el error, al estimar con la muestra validación, aunque se dan errores absolutos máximos de 30 μg de Chl-a/l, la mayor parte de la diferencia positiva se produce en los valles. Es decir, el modelo



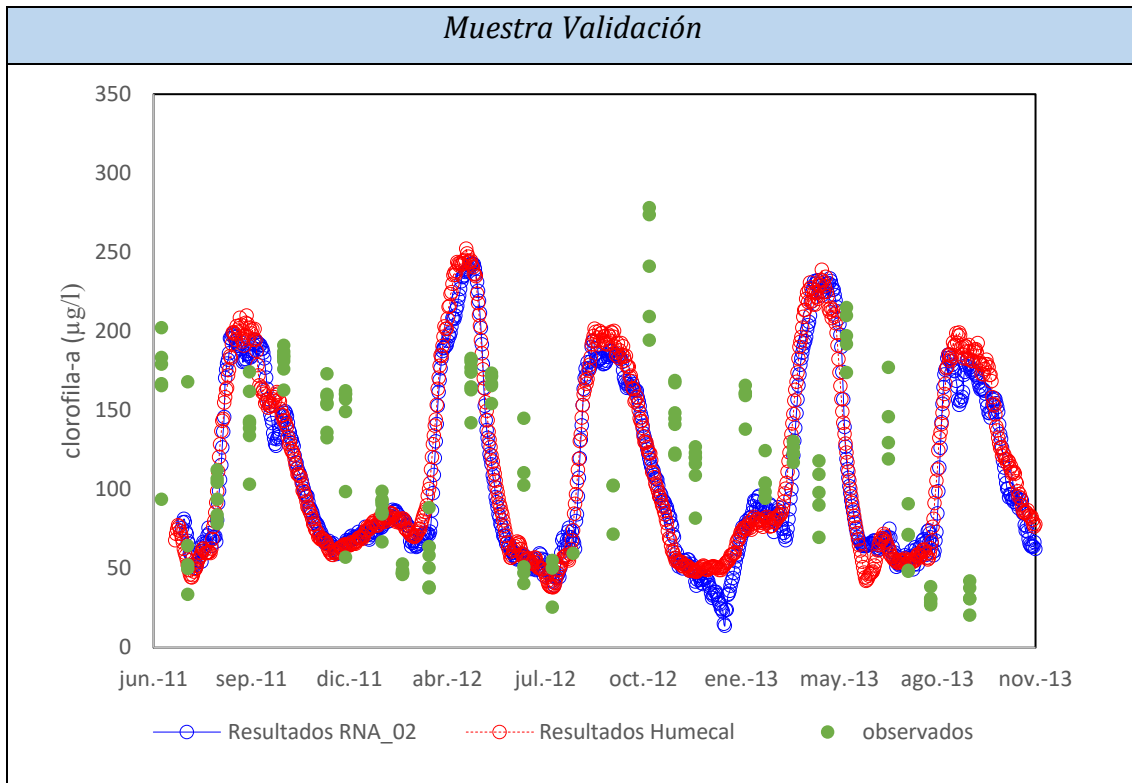
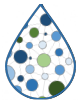
infravalora los resultados de Humecal cuando el sistema se encuentra en una situación de poca producción primaria.

- **Azul:** Resultados RNA-Escenario_02
- **Rojo:** Resultados Humecal



Gráfica 12 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_02

Se observa que los resultados de la RNA se ajustan mejor en los descensos de clorofila-a que en los incrementos. También tiene un mejor comportamiento cara a estimar la proliferación en el mes de septiembre (segundo pico en la Muestra Validación) que el mes de abril (primer pico, Muestra Validación). En cambio, el modelo sobreestima los valores máximos de abril y subestima los máximos de septiembre. Ante un caso real siempre es preferible un modelo que se situó del lado de la seguridad ambiental y no infravalore la variable objetivo.



Gráfica 13 Representación datos observados, resultados Humecal y resultados RNA_02

Una vez ejecutada la red se ha representado en la gráfica 12 los datos de clorofila-*a* observados en la laguna durante el periodo 2011-13, junto con los resultados del modelo Humecal y la estimación de la RNA_02.

Teniendo en cuenta la calidad de los resultados del modelo, se ha considerado innecesaria la aplicación en este escenario de una red del tipo recurrente.

5.5.3. Escenario_03

El planteamiento de este escenario sigue la línea del anterior. El objetivo principal es proveer a los gestores una idea de la tendencia que va a seguir la concentración de clorofila-*a* durante los siguientes 15 días. Para ello se ha forzado un desfase del periodo mencionado entre los datos de entrada y los de salida.

Una vez preparados manualmente el archivo de entrada en formato “.csv”, se han cargado los datos en el código y se ha ejecutado para seleccionar la estructura que ofrece un mejor ajuste. A diferencia con la red del Escenario 1, en este caso el modelo precisa de mucha información para poder representar la relación entre las concentraciones de los inputs en *t* y la concentración de clorofila-*a* en *t*+15.

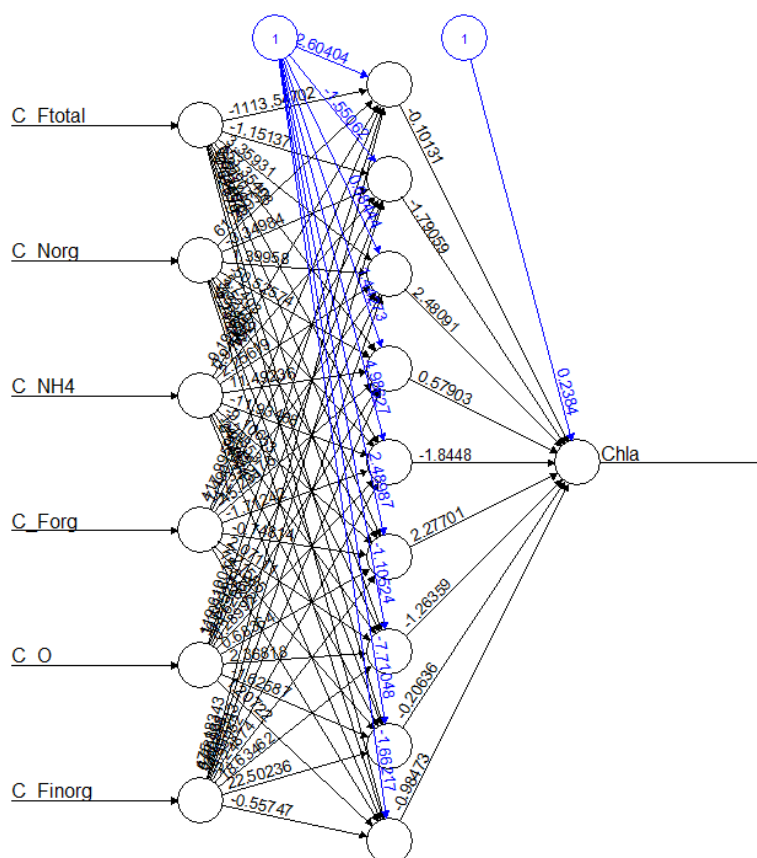


Ilustración 10 Red Neuronal Artificial- Escenario_03

Como en el modelo desarrollado en el Escenario_02 la red ya perdía ajuste, se ha intentado proveer a este modelo de mayor flexibilidad mediante el aumento progresivo del número de neuronas. Pero el aumento de neuronas ocultas impide que la red generalice durante la fase de validación, produciéndose una reducción en el ajuste global. Tras evaluar los resultados y comparar los modelos generados, se concluye que vuelve a ser la red de seis inputs y nueve neuronas oculta la que tiene la mejor respuesta. Ya que aumentando el número de neuronas no se mejora el resultado.

En la tabla 8, se muestran los pesos sinápticos de la red una vez superada la fase de entrenamiento. Para evitar mínimos locales, se ha lanzado el algoritmo de entrenamiento 100 veces. El punto de inicio del algoritmo es aleatorio por lo que es poco probable que se den dos simulaciones completamente iguales.

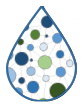
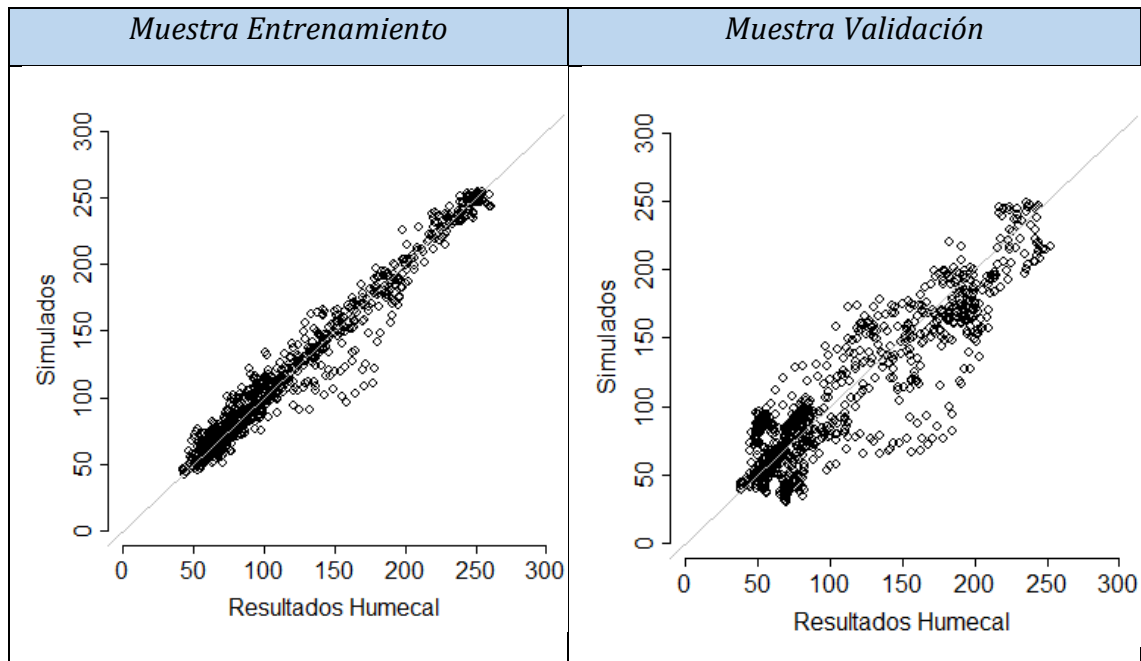


Tabla 9 Pesos de la RNA-Escenario_03

		Neuronas ocultas								
		1º	2º	3º	4º	5º	6º	7º	8º	9º
Pesos, 1º CAPA	C_Ftotal	-1113.55	-1.15	3.36	153.35	6.27	7.51	-27.24	-0.61	-7.33
	C_Norg	61.30	-3.35	1.40	-50.53	5.74	3.52	7.53	-6.19	-4.25
	C_NH4	-9.19	0.92	2.26	11.49	-11.93	-9.11	1.30	-24.94	-0.55
	C_Forg	418.00	7.49	-4.23	-45.29	-1.71	-0.75	2.07	7.22	15.88
	C_O	119.61	1.10	-0.58	-14.63	0.29	0.68	2.37	-1.63	1.21
	C_Finorg	475.18	8.29	0.87	-30.63	-6.89	-5.25	18.63	22.50	-0.56
	Bias 1º	-32.60	-1.55	0.68	1.44	4.99	2.49	-1.11	-7.71	-1.66
Pesos, 2º CAPA		-0.10	-1.79	2.48	0.58	-1.84	2.28	-1.26	-0.21	-0.98
Bias 2º		0.24								

Al igual que en el primer y segundo escenario el orden de inclusión de los *inputs* está determinado por el valor de su índice de correlación de Pearson. El cual aporta información de la correlación lineal entre cada uno de los *inputs* y el *output*. Las variables de entrada a la red son: fósforo total, fósforo orgánico, fósforo inorgánico, nitrógeno orgánico, amonio y el oxígeno disuelto.



Gráfica 14 Ajuste de la RNA-Escenario_03

A medida que se aumenta el desfase entre el *input* y el *output*, la relación directa entre ambas variables decrece. La RNA ajustan su estructura para representar dicha relación, pero a medida que esta disminuye la red pierde capacidad de adaptación.

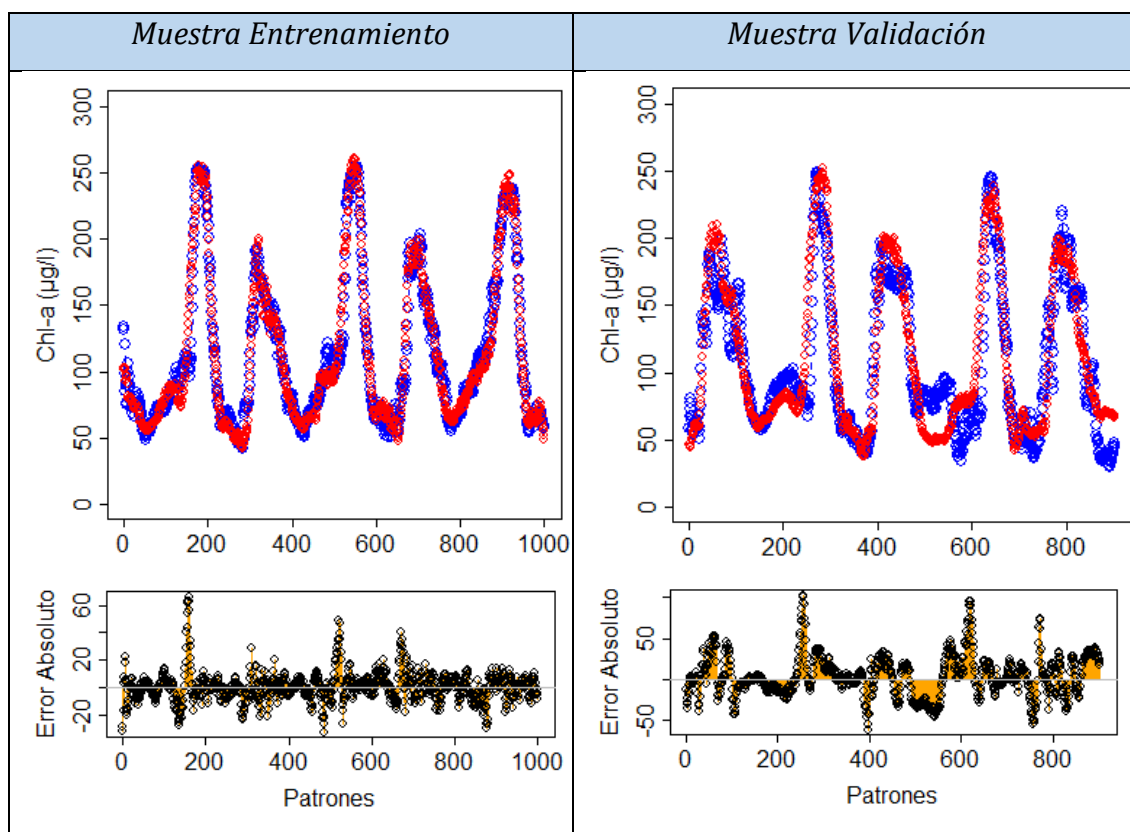


Es decir, las RNAs no son capaces de estimar la variable objetivo, si entre los *inputs* y los *outputs* no existe una dependencia significativa.

Aun así, aunque se pierde ajuste en la simulación de la muestra validación respecto a la simulación con la muestra entrenamiento, en la fase de validación se sigue obteniendo buenos resultados con un r^2 de 0.816.

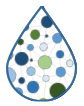
Respecto al rango en el que se distribuye el error, al estimar con la muestra validación, se dan errores absolutos máximos de $-100\mu\text{g}$ de Chl-*a*/l y el RMSE ha aumentado hasta $25\mu\text{g}$ de Chl-*a*/l.

- **Azul:** Resultados RNA-Escenario_03
- **Rojo:** Resultados Humecal

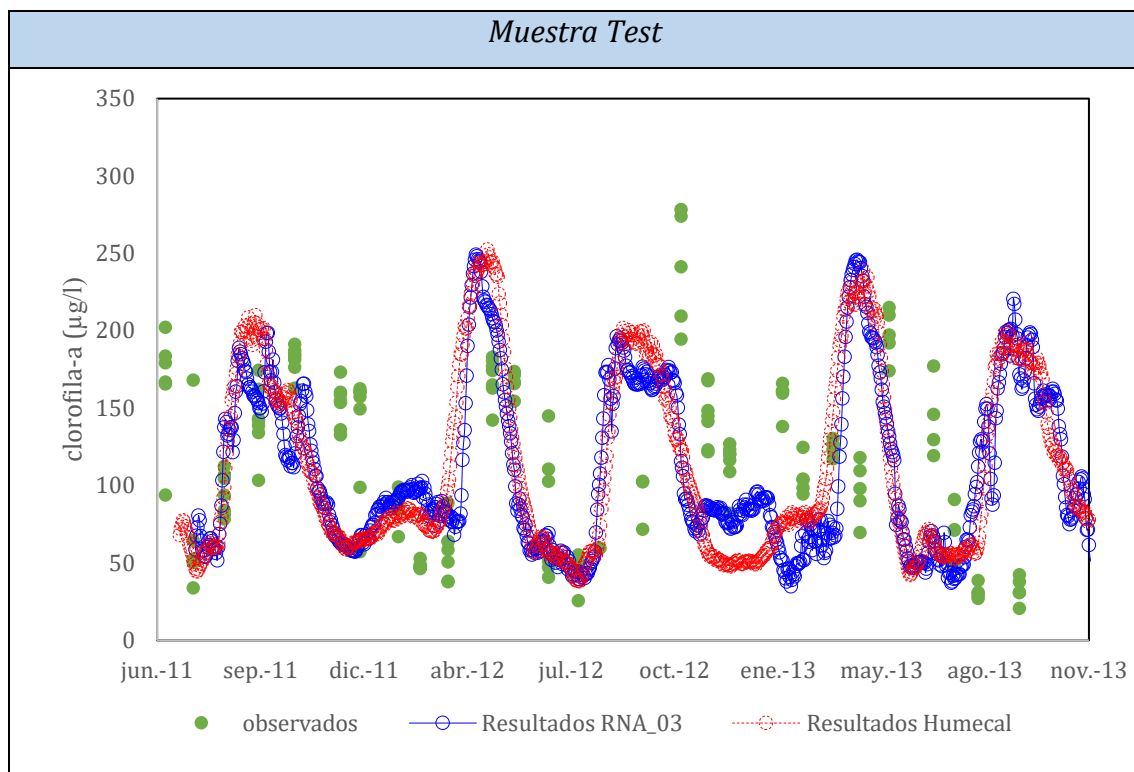


Gráfica 15 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_03

La red no es capaz de identificar cuando un aumento puntual acaba en un pico máximo o cuando el crecimiento poblacional está limitado por otros factores no incluidos como *inputs*.



Dos parámetros que afectan directamente a la producción primaria de biomasa algal, son la temperatura y la radiación. Por ejemplo, si en invierno aumenta la carga de entrada de nutrientes la comunidad de microalgas puede sufrir un pequeño incremento puntual, pero las bajas temperaturas y la limitada radiación incidente impedirán que se produzca un *bloom*. Al no incluir ninguna variable física como *input*, la red no responde bien a los incrementos puntuales durante las estaciones más frías y genera oscilaciones erróneas.



Gráfica 16 Representación datos observados, resultados Humecal y resultados RNA_03 de la muestra de validación

Al igual que en el Escenario_02, se observa que los resultados de la RNA se ajustan mejor en los descensos de clorofila-*a* que en los incrementos. En cambio, la red del Escenario_03 tiene un mejor comportamiento cara a estimar la proliferación en el mes de abril que el mes de septiembre.

Teniendo en cuenta las limitaciones de la RNA_03 y la calidad de los resultados, se ha considerado innecesaria la aplicación en este escenario de una red del tipo recurrente.

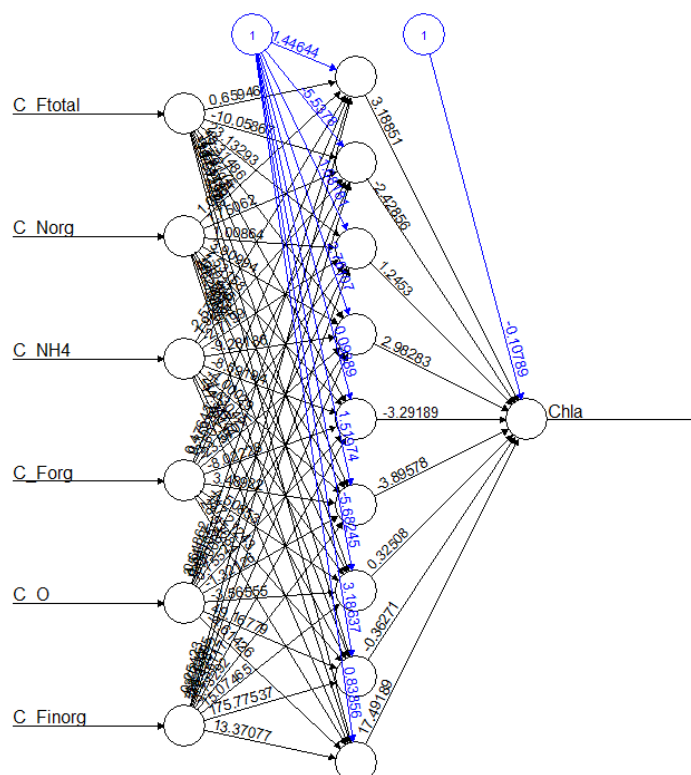


5.5.4 Escenario_04

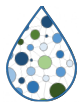
En el último de los cuatro escenarios se ha aumentado el desfase hasta 30 días entre los valores *inputs* y el *output*. Una red de estas características podría ser una herramienta útil para los gestores del parque, ya que las simulaciones extraídas podrían aclarar las necesidades hídricas de la laguna.

La Confederación Hidrográfica del Júcar es organismo responsable de la asignación de las entradas de agua al paraje. Existen alternativas de actuación frente a al control de la proliferación masiva de microalgas relacionadas con los flujos de entrada de agua, conocidos como *flushings*.

Los *flushings* consisten en lavados inteligentes que se inician cuando las concentraciones de clorofila-a están creciendo y alcanzan un determinado valor. Si los agentes responsables en la asignación de agua pudiesen estimar la evolución de la clorofila-a, con el margen establecido en el presente escenario, la aplicación de dicha alternativa podría ser una buena opción para gestionar el problema de la eutrofización en la Albufera.



Gráfica 17 Red Neuronal Artificial- Escenario_04

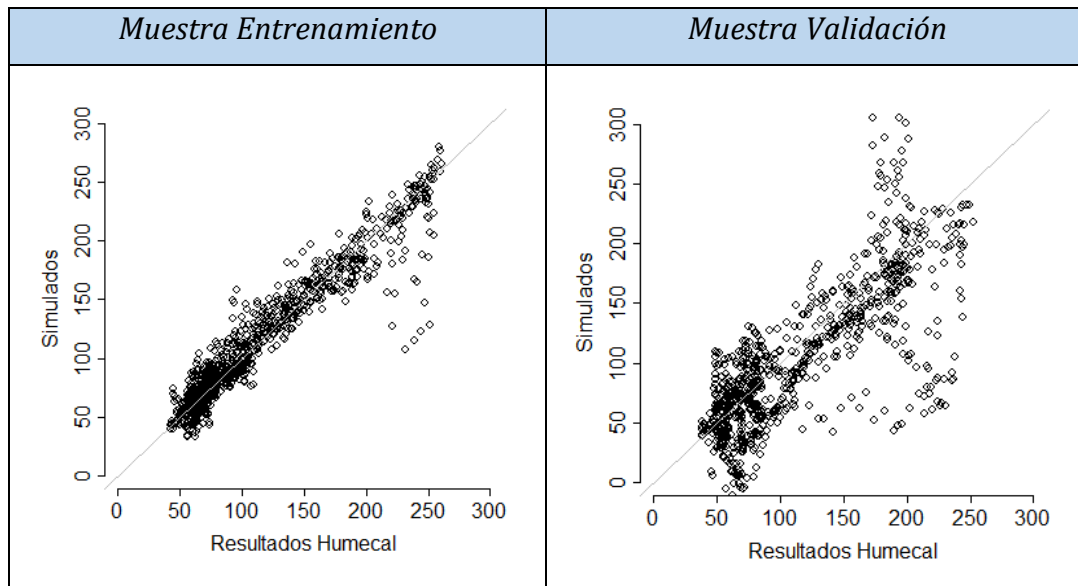


La estructura de la red generada es la misma que las del escenario 2 y 3, seis neuronas de entrada, nueve neuronas ocultas y una neurona de salida. A continuación, se muestran los pesos tras la ejecución de la fase de entrenamiento.

Tabla 10 Pesos de la RNA-Escenario_04

		Neuronas ocultas								
		1º	2º	3º	4º	5º	6º	7º	8º	9º
Pesos, 1º CAPA	C_Ftotal	0.66	-10.06	23.13	48.31	17.22	-7.29	-13.74	-334.33	-19.97
	C_Norg	1.07	1.75	1.01	3.91	7.53	1.19	-28.19	25.25	-16.32
	C_NH4	-2.58	1.93	-2.72	-9.28	-8.59	-4	27.63	-54.34	-17.7
	C_Forg	0.48	9.36	-3.6	-23.57	-8.02	3.49	-4.51	187.21	28.93
	C_O	-0.65	3.54	0.8	-3.7	0.76	-1.32	-3.57	49.17	-4.61
	C_Finorg	-0.25	4.39	-18.4	-25.36	-12.36	3.73	15.07	175.78	13.37
Bias 1º		1.45	-5.54	-1.68	2.76	0.09	1.52	-5.68	3.19	0.84
Pesos, 2º CAPA		3.19	-2.43	1.25	2.98	-3.29	-3.9	0.33	-0.36	17.49
Bias 2º		-0.11								

Tras la optimización de los pesos sinápticos, se ha ejecutado el modelo con las dos muestras de datos y se han graficado para evaluar la calidad de la simulación. En este caso se observa un empeoramiento de los resultados obtenidos con la muestra de validación respecto a la muestra entrenamiento. El r^2 de la modelación en la muestra validación es de 0.612 y el RMSE ha aumentado hasta 40µg de chl-a/l.



Gráfica 18 Ajuste de la RNA-Escenario_04

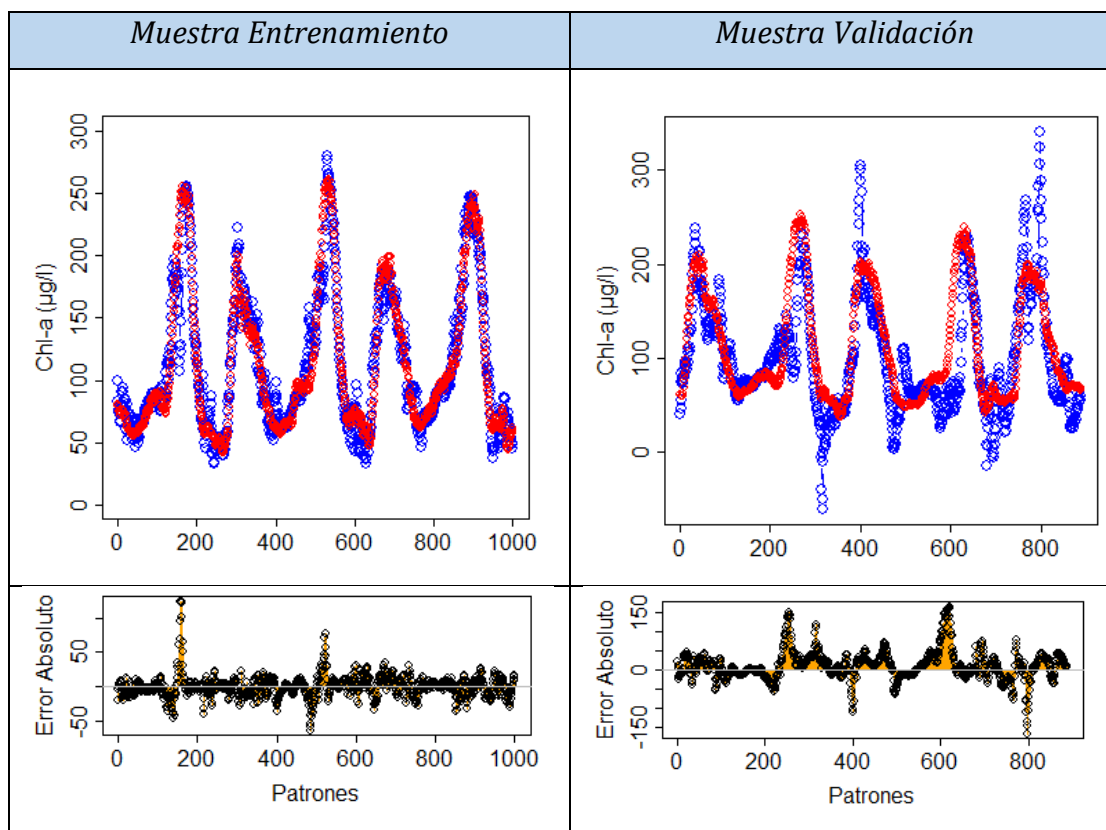
Al dispersarse, mayoritariamente la nube de puntos en la parte inferior a la recta de 45º, se asume que el modelo está infravalorando la concentración de clorofila-a en



la laguna. Aunque aparentemente podría tratarse de un caso de sobreentrenamiento, se ha comprobado que si se reducen el número de neuronas ocultas la red no es capaz de generalizar, es decir la red no responde.

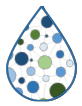
Esta reducción en el índice de correlación puede estar ligado a la desconexión entre las variables de entrada en t y el *output* en $t+30$. Probablemente si se siguiese aumentando el desfase alcanzaríamos un punto en el que no existiese relación entre los *inputs* y el *output*.

- **Azul:** Resultados RNA-Escenario_04
- **Rojo:** Resultados Humecal

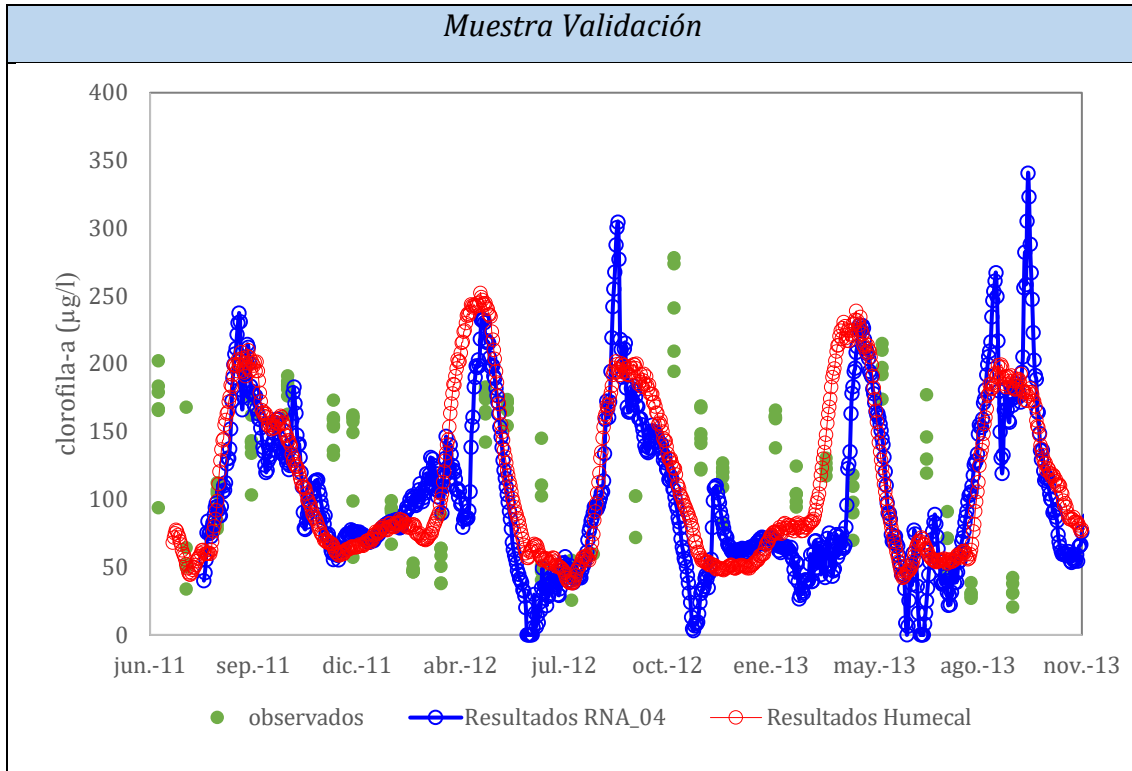


Gráfica 19 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_04

Al graficar los resultados como series temporales, se comprueba lo afirmado en el párrafo anterior. La simulación con la muestra de validación infravalora el aumento de clorofila-*a* en el mes de septiembre, llegando a haber una diferencia de $150\mu\text{g}$ de chl-*a*/l entre los resultados de Humecal y la RNA.



Además, los resultados sufren notables oscilaciones como en el tercer escenario. No obstante, si evaluamos la estimación del pico de abril, teniendo en cuenta las limitaciones de la RNA, la respuesta de la red se ajusta bastante bien a los resultados de Humecal.



Gráfica 20 Representación datos observados, resultados Humcal y resultados RNA_04 de la muestra de validación

En la gráfica 20, no se han graficado los resultados negativos de la RNA_04, ya que no existen concentraciones negativas de clorofila-a en la naturaleza.

En este caso si se ha optado por incorporar a la estructura ya diseñada una conexión recurrente que redirija el *output* estimado a una neurona de entrada. El dato estimado se incluirá como *input* para la estimación del *output* en $t+1$.

5.5.4.1 Red Recurrente Escenario_04

El objetivo de este apartado no es centrarse en el diseño de una red *Jordan* que mejor se ajuste, si no plantear el uso de una tipología menos común para resolver problemas como el abordado en el presente TFM. En principio, al incluir a la red información de la situación del sistema en $(t-1)$, se espera una mejorar el ajuste y se debería reducir las oscilaciones entre patrones sucesivos.



Para ello se ha incluido la variable clorofila-*a* en (*t-1*), es decir, si el modelo quiere predecir la clorofila-*a* con 30 días de antelación a la red entraran los *inputs* en (*t*) y el resultado anterior o (*t+29*). Dicha información se incorpora a partir de una conexión recurrente entre la neurona de salida y una nueva neurona de entrada. En la ilustración 12 se puede observar la estructura de la red construida. Mediante esta nueva conexión se evitan oscilaciones bruscas y, por tanto, los aumentos o descensos sufridos por la concentración de clorofila-*a* en el caso de estudio se darán de forma progresiva.

En el software R existen paquetes específicos para implementar tipologías muy concretas de redes neuronales. Para el uso de una estructura neuronal del tipo *Jordan*, se ha empleado el paquete RSNSS. En comparación con el paquete Neuralnet, el entrenamiento de redes *Jordan* precisan de tiempos computacionales relativamente altos.

No se ha ejecutado la parte del código específica para la elección de la estructura neuronal, se ha decidido emplear una estructura neuronal igual a la del segundo, tercer y cuarto escenario. Con una única diferencia, al incluirse otro input por la conexión recurrente esta red constará de siete neuronas de entrada, una más que en los casos anteriores.

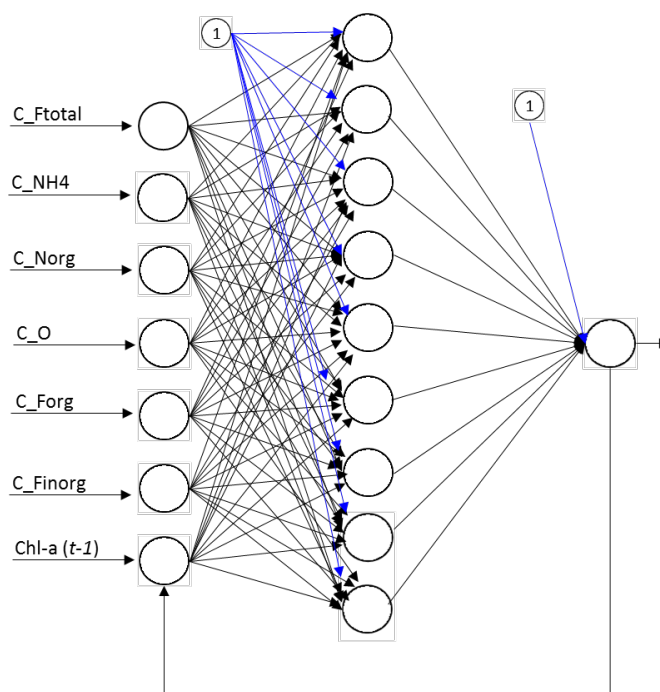
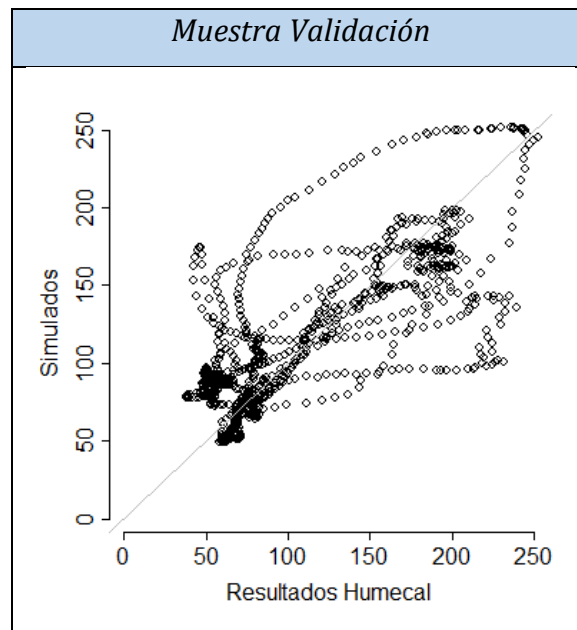


Ilustración 11 Red Neurona Artificial Jordan- Escenario_04.



Como ya se ha comentado, en un ecosistema natural el crecimiento de las especies del tipo r, conocidas por tener crecimientos exponenciales de su población, acaba estando limitado por la competencia por los recursos. En el caso de las microalgas, el espacio y la disponibilidad de nutrientes acaba frenando la proliferación. La red recurrente al tener en cuenta la concentración de clorofila-a en $(t-1)$ incluye dicha limitación natural.

En la gráfica 20, se observa como la nube de punto se dispersa por el gráfico como en el caso del Escenario_04. El ajuste del modelo alcanza un r^2 de 0.542, el RMSE es de 40 μg de Chl-a/l y el error máximo es de $\pm 100\mu\text{g}$ de Chl-a/l. Aunque el error cuadrático medio no aumenta, si comparamos la respuesta con los resultados de Humecal se ha perdido calidad en la estimación.



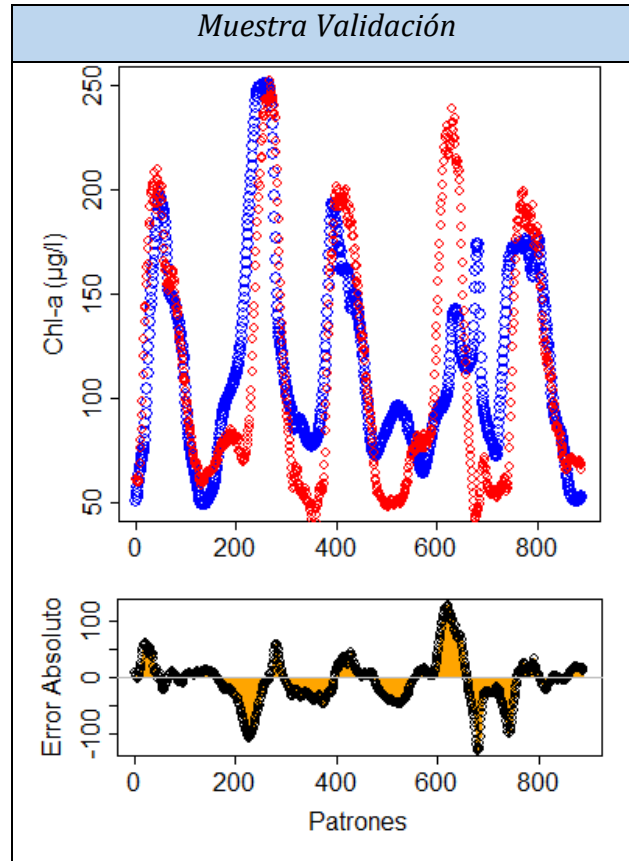
Gráfica 21 Ajuste de la RNA-Escenario_04

En términos de ajuste al modelo Humecal el uso de una red semirecurrente para la modelación de la clorofila-a en el Escenario_04, empeora los resultados respecto al uso de una tipología MLP.

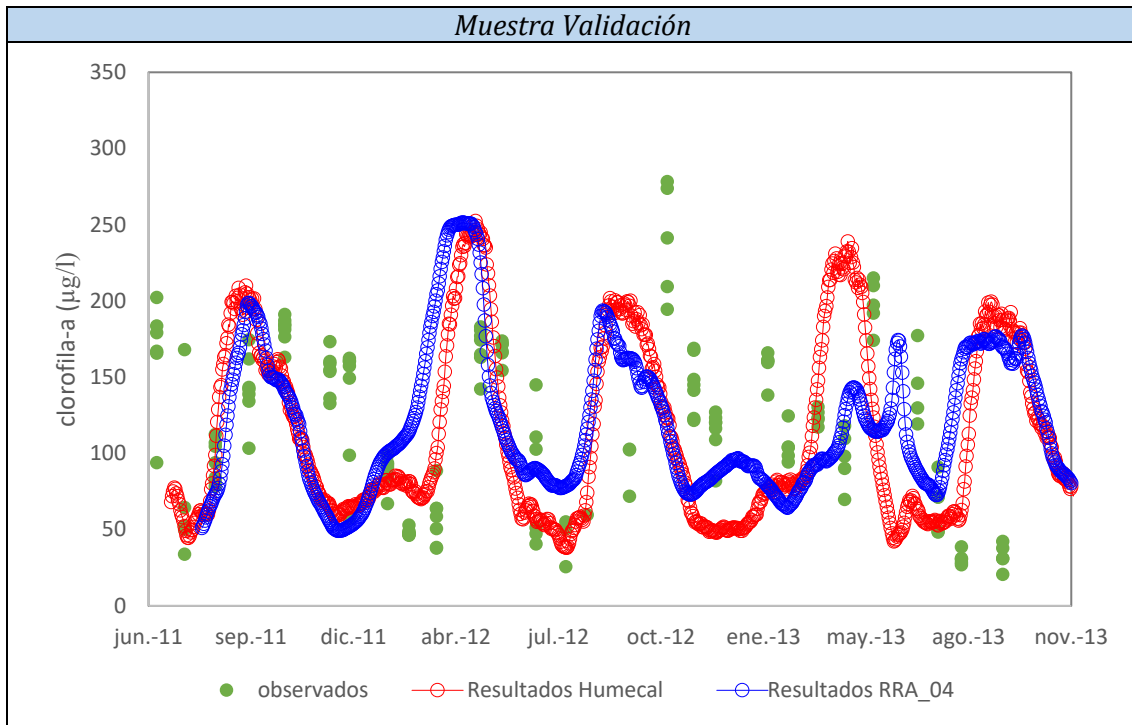
Pero si se observa el gráfico 21, se ha resuelto el problema de las fluctuaciones. En la realidad los cambios sufridos por los parámetros de calidad de agua no suelen producirse de forma repentina, si no están sujetos a procesos físicos-químicos o biológicos con los que se producen incrementos positivos o negativos de forma



progresiva. Por lo que, aunque se obtengan buenos resultados en el segundo y tercer escenario con redes del tipo MLP, mediante la implementación de redes del tipo *Jordan* se puede resolver el problema de la inestabilidad en los resultados.



Gráfica 22 Series estimadas de clorofila-a del modelo Humecal y la RNA-Escenario_05



Gráfica 23 Representación datos observados, resultados Humecal y resultados RR_04

Por último, al graficar los resultados de los dos modelos con los observados de clorofila-*a* en la laguna, se repara en que ambas estimaciones representan una tendencia similar. Pero la red sobreestima las concentraciones más bajas durante la estación de invernada y estival, y no responde al incremento de las concentraciones de clorofila-*a* durante abril del 2013.

El ajuste emporea respecto al modelo MLP para este Escenario. Pero teniendo en cuenta, que no se ha identificado la estructura neuronal que pudiese dar mejor respuesta, los resultados no son concluyentes desde el punto de la capacidad de esta tipología neuronal. En cambio, si se ha observado una mejora notable en la estabilidad de las respuestas. Por esta razón, aunque los tiempos computacionales sean mayores, se concluye que esta tipología puede ser muy útil a la hora de resolver problemas relacionados con series temporales.

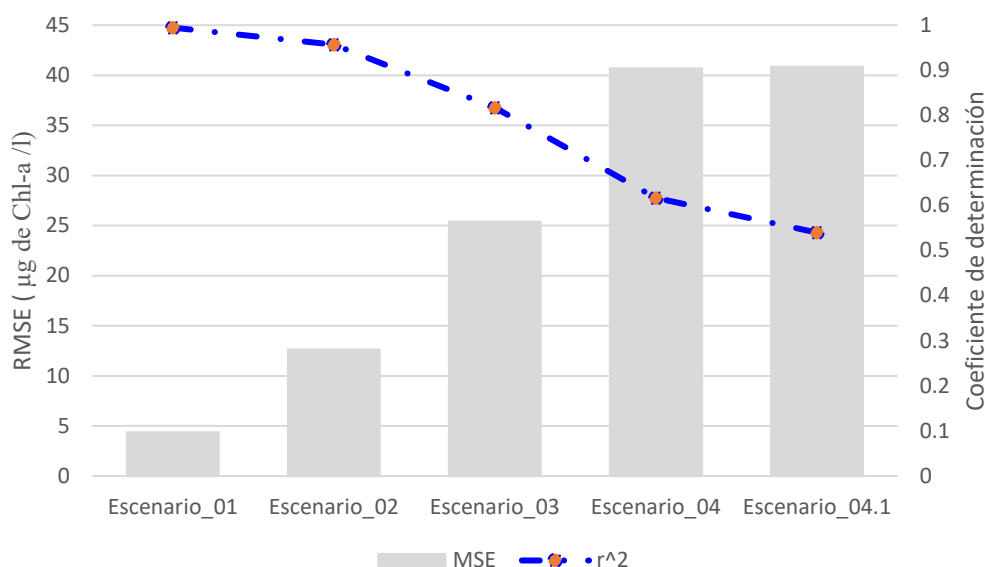
5.5.5 Comparación escenarios

Cada sistema tiene una inercia, es decir la situación en el momento *t* afectará más o menos a los futuros escenarios. Aplicado al caso de estudio, la evolución futura de la concentración de clorofila-*a* se verá condicionada por la situación del ecosistema en *t*.



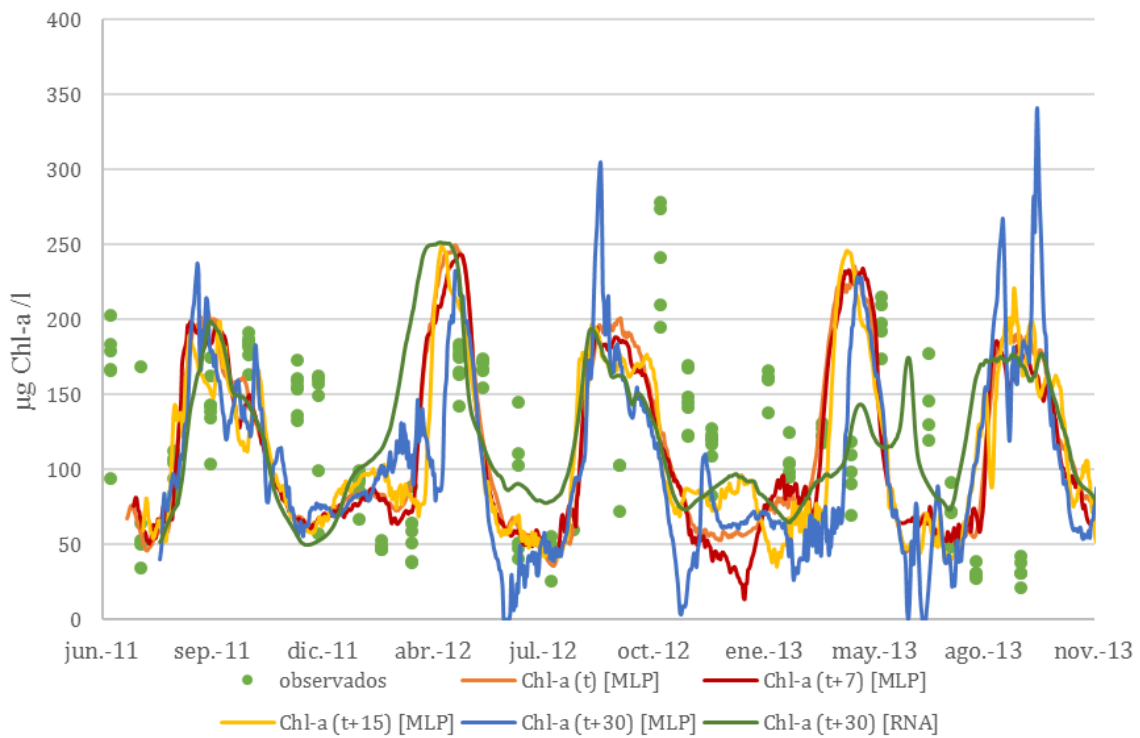
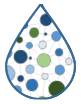
Tras la ejecución de los cuatro escenarios y las cinco redes presentadas, se puede observar una tendencia descendente en los índices de r^2 y una ascendente en el caso del error cuadrático medio (Gráfica 24). En los tres primeros escenarios, las redes generadas responden correctamente y dan buenos resultados. Pero entre el ajuste de la red del Escenario_03 y la del Escenario_04 se produce una reducción muy significativa de la calidad del modelo. La cuarta red es incapaz de representar un 40% de los resultados de Humecal.

Si se compara con las tres primeras, que son capaces de simular entre el 80-99% de los datos, los resultados del último escenario no son buenos. En cambio, si se tiene en cuenta el desfase entre las variables *input* y la variable *output*, el modelo podría ser útil en la toma de decisiones sobre el PNAV a medio plazo.



Gráfica 24 RMSE y r^2 de las cinco RNA

La reducción progresiva en la calidad de los resultados puede tener relación con la desconexión entre la situación del ecosistema en t y la situación en $t+30$. La complejidad de los ecosistemas naturales los hace tremendamente dinámicos y como demuestran los resultados la modelación de variables biológicas a largo plazo se complica a medida que aumentamos el desfase.



Gráfica 25 Resultados de los cuatro escenarios y datos observados de clorofila-a

Por último, mediante la red *Jordan*, diseñada para modelar la clorofila-a en el Escenario_04, se demuestra la capacidad de dicha tipología para estimar variables en serie temporales. Si se comparan los resultados de los modelos basados en *MLP*, con las respuestas de la red *Jordan*, se observa una reducción significativa de las oscilaciones en la estimación. La inestabilidad de los resultados de los modelos *MLP* están asociados a la desconexión entre el *output* en (*t*) y el *output* en (*t+1*). El uso de redes *Jordan* puede ser una buena alternativa para evitar dichas fluctuaciones, pero es necesario realizar un número mayor de pruebas para demostrarlo.



6. Conclusiones

En este apartado de la memoria se recopilan las conclusiones alcanzadas tras el desarrollo de cinco RNAs, para la modelación de las concentraciones de clorofila-*a* en la laguna del Parque Natural de la Albufera, en cuatro escenarios:

- Se han desarrollado cinco modelos para la simulación de la concentración de clorofila-*a* en la Albufera de Valencia. Cuatro de ellos de la tipología *Multilayer Perceptron* y uno del tipo *Jordan* o recurrente simple. Los modelos han conseguido ajustes con un valor del r^2 de hasta 0.994. De forma general la modelación desarrollada en este TFM consigue representar los resultados del modelo mecanicista.
- La red desarrollada en el Escenario_01, supone una simplificación del modelo Humecal. Facilita el uso de este tipo de modelos por parte de los gestores y reduce significativamente los parámetros necesarios para la modelación de la clorofila-*a* en la laguna sin perder ajuste.
- Los resultados obtenidos en los Escenarios 2, 3 y 4 aportan información sobre la inercia del sistema. Reflejando una desconexión progresiva a medida que se va aumentando el desfase temporal. Pero teniendo en cuenta condiciones con las que se entrenan, las RNAs permiten estimar la tendencia futura de la clorofila-*a* con un r^2 de hasta 0.957
- El software R ha resultado ser una buena herramienta para el desarrollo del presente TFM. La versatilidad y la gran variedad de paquetes programados para cubrir las necesidades del proyecto junto con la existencia de numerosas referencias y casos de estudio han ayudado a agilizar el proceso de aprendizaje. Además, este software ofrece una serie de paquetes específicos para la implementación de RNAs bien consolidados en el ámbito de estudio de estas técnicas.
- Mediante la consulta de bibliografía y el desarrollo del TFM se ha comprobado la capacidad de ajuste de estas técnicas y el buen comportamiento ante problemas como el que se plantea en este documento. Aunque la baja disponibilidad de datos observados limita su uso, las RNAs se van consolidando en el ámbito de la modelación de calidad de agua.



- La falta de muestras de datos adecuadas y suficientemente grandes para entrenar las RNAs obliga a buscar alternativas como la empleada en este TFM. Aunque desde un primer momento se asume una simplificación de la realidad por no emplear datos observados. El uso de resultados de modelos más complejos, como Humecal, para la fase de entrenamiento supone una alternativa a la falta de datos.
- Al igual que ocurre con los modelos mecanicistas, con las RNAs es aconsejable realizar actualizaciones y nuevas calibraciones para que las simulaciones se ajusten en la medida de lo posible a la situación del sistema real. Además, en el caso presentado, mientras no se disponga de suficientes datos observados para calibrar el modelo, dichas actualizaciones son la mejor forma de incorporar en la red la realidad del ecosistema modelado.
- Uno de los mayores problemas localizados durante el desarrollo de este TFM, ha sido que la bibliografía consultada no especifica la mejor forma para la elección de la estructura de la red. En la mayoría de los documentos optan por el método prueba-error o por decisión de personal experto. La metodología presentada facilita la elección de la estructura de la RNA, ya que permite automatizar las sucesivas pruebas y guardar los resultados para poder compararlos fácilmente.
- En la Albufera, el patrón anual de la concentración de clorofila-*a* se caracteriza por tener dos máximos, uno en abril y otro en septiembre. Dependiendo de la época del año, las condiciones ambientales y la situación del sistema son diferentes antes de iniciar la proliferación. Frente a un patrón de entrada similar (*inputs*), se produce un comportamiento distinto (*output*). Es decir, las variables empleadas como *inputs* no son las únicas que influyen sobre las variaciones anuales sufridas por la clorofila-*a* en la Albufera. Por lo que las redes son incapaces de representar dos comportamientos frente a una misma situación. De ahí que los modelos en los Escenarios 2, 3 y 4 representen mejor un pico que otro.



- A medida que se aumenta el desfase entre el *input* y el *output*, la relación directa entre ambas variables decrece. La RNA ajustan su estructura para representar dicha relación, pero a medida que esta disminuye la red pierde capacidad de adaptación. Es decir, las RNAs no son capaces de estimar la variable objetivo, si entre los *inputs* y los *outputs* no existe una dependencia clara.
- Cada sistema tiene una inercia, es decir la situación en el momento t afectará más o menos a los futuros escenarios. Aplicado al caso de estudio, la evolución futura de la concentración de clorofila- a se verá condicionada por el estado del ecosistema en t , pero con límites. La complejidad de los ecosistemas naturales los hace tremendamente dinámicos. Los resultados demuestran que la modelación de variables biológicas a largo plazo se complica a medida que aumentamos el desfase.
- Las redes de los Escenarios 02, 03 y 04, no son capaces de representar bien los pequeños crecimientos puntuales de clorofila- a durante las estaciones frías. No disponen de información sobre las limitaciones ambientales en dichas épocas, debido a que no se ha incluido ninguna variable de entrada física.
- En el Escenario_04, el ajuste de la red *Jordan* emporo respecto al modelo MLP, de un r^2 de 0.612 (MLP) a 0.542 (*Jordan*). Pero teniendo en cuenta, que no se ha identificado la estructura neuronal que pudiese dar mejor respuesta, los resultados no son concluyentes desde el punto de la capacidad de esta tipología. En cambio, si se ha observado una mejora notable en la estabilidad de las respuestas de la red. Por lo que, aunque los tiempos computacionales sean mayores, se concluye que esta tipología puede ser muy útil a la hora de resolver problemas relacionados con series temporales.
- Mediante la ejecución conjunta de las 4 redes con la misma muestra de entrada, se podría obtener una estimación a corto y medio plazo de la tendencia de las concentraciones de clorofila- a en la Albufera.

Como conclusión final, los resultados obtenidos demuestran la capacidad de las RNA para ajustar su estructura y representar variables tan complejas como la clorofila- a .



6. Conclusiones

A pesar de la falta de datos observados se ha podido validar el modelo con los resultados del modelo mecanicista Humecal. La inclusión de nuevas simulaciones de la Albufera, con la herramienta Humecal, en el plan especial del PNAV evidencia la calidad del modelo del que se ha partido. La mayor diferencia entre dicho modelo y las RNA es el número de parámetros que necesitan para obtener resultados similares (Humecal- Escenario_01). La simplificación de los modelos facilita su uso por parte de los gestores responsables, por lo que se considera que el proyecto ha alcanzado los objetivos propuestos.



7. Futuras líneas de investigación

En este apartado se exponen las líneas de investigación que se generan a partir del TFM descrito en esta memoria. Frente a un amplio abanico de posibilidades, queda demostrada la polivalencia de las RNAs para modelar diferentes parámetros en el ámbito de los recursos hídricos. Se propone ampliar los conocimientos al respecto mediante el desarrollo de los siguientes campos:

- Aplicar RNAs para modelar otros contaminantes y comprobar si son capaces de resolver satisfactoriamente problemas relacionados con otros parámetros de calidad del agua.
- Incluir variables físicas de datos observados y no patrones anuales, para incluir en la red el efecto de la época del año sobre la evolución de la población de microalgas en la laguna.
- Recopilar datos observados para comprobar si el modelo responde bien ante ellos y si no es así, reentrenar las redes con datos observados. Ya que esa ha sido la mayor limitación en el desarrollo del proyecto.
- Realizar un mayor número de pruebas con redes neuronales recurrentes del tipo *Jordan* para la modelación de sistemas ecológicos. Aplicándolas a nuevos escenarios y nuevas variables.
- Plantear el uso de estas técnicas para modelar otros casos de estudio. Como no se trata de un modelo mecanicista, si se desease aplicar las redes en otro escenario sería imprescindible recalibrar la estructura. Ya que el entrenamiento presentado no serviría para estimar la clorofila-*a* en otro ecosistema.
- Diseñar redes específicas para cada máximo de clorofila-*a*, de esta forma se evitaría una generalización excesiva. Como se ha ido observado dependiendo del escenario, la red se ha ajustado mejor al pico de abril o al pico de septiembre. Aunque las concentraciones máximas sean similares, la situación del sistema no lo es. Para ello se podrían desarrollar dos redes de 6 meses cada una o cuatro una por cada estación.
- Por último, programar una interfaz más amigable que permita al usuario ejecutar el código sin necesidad de tener nociones de su funcionamiento. El software R ofrece una aplicación para desarrollar aplicaciones web de



7. Futuras líneas de investigación

forma sencilla que permite a los usuarios interactuar con sus datos sin tener que manipular el código.



8. Referencias

- Abyaneh, H. Z. (2014). Evaluation of multivariate linear regression and artificial neural networks in prediction of water quality parameters. *JOURNAL OF ENVIRONMENTAL HEALTH SCIENCE & ENGINEERING*. <https://doi.org/10.1186/2052-336X-12-40>
- Andreu, J., Capilla, J., & Sanchís, E. (1996). AQUATOOL, a generalized decision-support system for water-resources planning and operational management. *Journal of Hydrology*, 177(3-4), 269-291. [https://doi.org/10.1016/0022-1694\(95\)02963-X](https://doi.org/10.1016/0022-1694(95)02963-X)
- Asce. (2000). ARTIFICIAL NEURAL NETWORKS IN HYDROLOGY. By the ASCE Task Committee on Application of Artificial Neural Networks in Hydrology 1. *Journal Of Hydrologic Engineering*, 5(April), 124-137. <https://doi.org/10.5121/ijsc.2012.3203>
- Belda, E. (2015). Desarrollo de herramientas para el análisis y modelación de calidad de aguas en humedales. Aplicación en el ámbito del Parque Natural de La Albufera de Valencia., 187.
- Benet Granell, J. . (1983). La Albufera de Valencia. Datos para una política de soluciones. *Revista de Obras Públicas*, 130(3211), 167-180.
- Blanco, C. (1974). Estudio de la contaminación de la albufera de Valencia y de los efectos de dicha contaminación sobre la fauna y la flora del lago. University of Valencia, Spain.
- Bravo-Oviedo, A., & Kindermann, G. (2004). Modelización de sistemas ecológicos mediante redes neuronales. *Actas de La Reunión de Modelización Forestal. Cuaderno de La Sociedad Española de Ciencias Forestales*, 316, 311-316.
- Camilo, J., & Rivera, O. (2002). MODELO ESTOCÁSTICO DE REDES NEURONALES PARA LA SÍNTESIS DE CAUDALES APLICADOS A LA GESTIÓN PROBABILÍSTICA DE SEQUÍAS. *Departamento de Ingeniería Hidráulica Y Medio Ambiente, UPV*.
- Carlson, R. E. (1977). A trophic state index for lakes. *Limnology and Oceanography*, 22, 361-369.



- Chapra, S. C. (1997). *Surface water-quality modeling*. Waveland press.
- Cho, S., Lim, B., Jung, J., Kim, S., Chae, H., Park, J., ... Park, J. K. (2014). Factors affecting algal blooms in a man-made lake and prediction using an artificial neural network. *Measurement*, 53, 224–233. <https://doi.org/10.1016/j.measurement.2014.03.044>
- Cover, T. M. (1969). Learning in pattern recognition. In *Methodologies of Pattern Recognition* (pp. 111–132). New York.
- Cybenko, G. (1988). Continuous valued neural networks with two hidden layers are sufficient. Tufts University.
- Dafauce Ruiz, C. (1975). La Albufera de Valencia un estudio piloto. Madrid: Ministerio de Agricultura.
- Desortová, B. (2007). Relationship between Chlorophyll-a concentration and Phytoplankton Biomass in Several Reservoirs in Czchoslovakia. *International Revue Der Gesamten Hydrobiologie Und Hydrographie*, 66(2), 153–169.
- Fausett, L. (1994). *Fundamentals of Neural Networks*. Englewood Cliffs: Prentice Hall.
- Ferriol-Gabarda, C. (2013). La eutrofización en los lagos someros mediterráneos: aplicabilidad de la DMA y un caso de estudio experimental en mesocosmos, 409. <https://doi.org/10.1017/CBO9781107415324.004>
- Funahashi, K. (1989). On the approximate realization of continuous mapping by neural. *Neural Networks*, 183–192.
- Gabriela Onandia Bieco. (2015). Ciclo del carbono y modelización biogeoquímica de un lago somero hipertrófico: la Albufera de Valencia.
- Gazzaz, N. M., Yusoff, M. K., Aris, A. Z., Juahir, H., & Ramli, M. F. (2012). Artificial neural network modeling of the water quality index for Kinta River (Malaysia) using water quality variables as predictors. *Marine Pollution Bulletin*. <https://doi.org/10.1016/j.marpolbul.2012.08.005>
- Gestal, M. (2009). Introducción a las Redes de Neuronas Artificiales, 1–20. Retrieved



from <http://sabia.tic.udc.es/~mgestal/cv/RNATutorial/TutorialRNA.pdf>

- Guallar, C., Delgado, M., Diogone, J., & Fernández-Tejedor, M. (2016). Artificial neural network approach to population dynamics of harmful algal blooms in Alfacs Bay (NW Mediterranean): Case studies of *Karlodinium* and *Pseudo-nitzschia*. *Ecological Modelling*, 338, 37–50. <https://doi.org/10.1016/j.ecolmodel.2016.07.009>
- Haykin, S. (1999). *Neural networks a comprehensive foundation*. New Jersey.
- Heddam, S. (2016). Multilayer perceptron neural network-based approach for modeling phyocyanin pigment concentrations: case study from lower Charles River buoy, USA. *Environmental Science and Pollution Research*, 23(17), 17210–17225. <https://doi.org/10.1007/s11356-016-6905-9>
- Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2), 251–257. [https://doi.org/10.1016/0893-6080\(91\)90009-T](https://doi.org/10.1016/0893-6080(91)90009-T)
- Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer Feed-forward Networks are Universal Approximators. *Neural Networks*, 359–366.
- Hoskins, J. C., & Himmelblau, D. M. (1992). Process control via artificial neural networks and reinforcement learning. *Computers & Chemical Engineering*, 16(4), 241–251. [https://doi.org/10.1016/0098-1354\(92\)80045-B](https://doi.org/10.1016/0098-1354(92)80045-B)
- Jang, J. S. R., & Sun, C. T. (1995). Neuro-Fuzzy Modeling and Control. *Proceedings of the IEEE*, 83(3), 378–406. <https://doi.org/10.1109/5.364486>
- Jordan, M. I. (1986). *Serial Order: A Parallel Distributed Processing Approach*.
- Lane, W. L. (1979). *Applied Stochastic Techniques (Last Computer Package)*. U.S: Division of Planning Technical Services.
- Lin, C.-T., & Lee, C. . G. (1996). *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*.
- Lu, F., Chen, Z., Liu, W., & Shao, H. (2016). Modeling chlorophyll-a concentrations using an artificial neural network for precisely eco-restoring lake basin.



- Ecological Engineering*, 95, 422–429.
<https://doi.org/10.1016/j.ecoleng.2016.06.072>
- Maier, H. R., & Dandy, G. C. (2000). Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environmental Modelling and Software*, 15(1), 101–124.
[https://doi.org/10.1016/S1364-8152\(99\)00007-9](https://doi.org/10.1016/S1364-8152(99)00007-9)
- Marín Diazaraque, J. M. (2007). Introducción a las redes neuronales aplicadas. *Manual Data Mining*, 1–31. Retrieved from halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/DM/tema3dm.pdf
- Martín, M., Oliver, N., Hernández-Crespo, C., Gargallo, S., & Regidor, M. C. (2013). The use of free water surface constructed wetland to treat the eutrophicated waters of lake L'Albufera de Valencia (Spain). *Ecological Engineering*, 50, 52–61.
<https://doi.org/10.1016/j.ecoleng.2012.04.029>
- Matich, D. J. (2001). Redes Neuronales: Conceptos Básicos y Aplicaciones. *Historia*, 55. Retrieved from <ftp://decsai.ugr.es/pub/usuarios/castro/Material-Redes-Neuronales/Libros/matich-redesneuronales.pdf>
- Mazzeo, N., Clemente, J., García-Rodríguez, F., Gorga, J., Kruk, C., Larrea, D., ... Scasso, F. (2002). Eutrofización: causas, consecuencias y manejo. *Perfil Ambiental Del Uruguay*.
- McCulloch, W. S., & Pitts, W. (1943). A Logical Calculus of the Idea Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, 5, 115–133.
<https://doi.org/10.1007/BF02478259>
- Miracle M.R., García, P., & Vicente, E. (1984). Heterogeneidad espacial de las comunidades fitoplactónicas de la Albufera de Valencia. *Limnética*, 1, 20–31.
- Najah, A., El-Shafie, A., Karim, O. A., & El-Shafie, A. H. (2013). Application of artificial neural networks for water quality prediction. *Neural Computing and Applications*. <https://doi.org/10.1007/s00521-012-0940-3>
- Najah, A., Elshafie, A., Karim, O. A., Jaffar, O., & Lecturer, S. (2009). Prediction of Johor River Water Quality Parameters Using Artificial Neural Networks. *European*



Journal of Scientific Research, 28(3), 1450–216.

- Obregón, N., Fragala, F., & Prada, L. (2003). Redes neuronales artificiales en hidroinformática. *Sistema*, (November), 1–5. Retrieved from <http://cinara.univalle.edu.co/archivos/pdf/61.pdf>
- Ochoa-Rivera, J. C. (2008). Prospecting droughts with stochastic artificial neural networks. *Journal of Hydrology*, 352(1–2), 174–180. <https://doi.org/10.1016/j.jhydrol.2008.01.006>
- Ochoa-Rivera, J. C., Andreu, J., & García-Bartual, R. (2007). Influence of Inflows Modeling on Management Simulation of Water Resources System. *Journal of Water Resources Planning and Management*, 133(2), 106–116. [https://doi.org/10.1061/\(ASCE\)0733-9496\(2007\)133:2\(106\)](https://doi.org/10.1061/(ASCE)0733-9496(2007)133:2(106))
- Paredes, J., & Lund, J. R. (2006). Refill and drawdown rules for parallel reservoirs: Quantity and quality. *Water Resources Management*, 20(3), 359–376. <https://doi.org/10.1007/s11269-006-0325-4>
- Pujol Reig, L. (2009). *Predicción de caudales en tiempo real en grandes cuencas utilizando Redes Neuronales Artificiales. Tesis doctoral.* <https://doi.org/10.4995/Thesis/10251/6422>
- Romo, S., & Miracle, M. R. (1994). Population dynamics and ecology of subdominant phytoplankton species in a shallow hypertrophic lake (Albufera of Valencia, Spain). *Hydrobiologia*, 273(1), 37–56. <https://doi.org/10.1007/BF00126767>
- Rumelhart, D. E., Smolensky, P., McClelland, J. L., & Hinton, G. E. (2013). Schemata and Sequential Thought Processes in PDP Models. *Readings in Cognitive Science: A Perspective from Psychology and Artificial Intelligence.* <https://doi.org/10.1016/B978-1-4832-1446-7.50020-0>
- Salazar, A. M., Rodríguez, G. J. M., & Cabrera-ríos, M. (2006). Caracterización Estadística y Optimización de Redes Neuronales Artificiales para Pronóstico de Series de Tiempo Pronóstico de un Solo Período. *Computación Y Sistemas*, 10, 69–81.
- Santín González, D., & Valiño Castro, A. (2003). *Eficiencia técnica y redes neuronales:*



- un modelo para el cálculo del valor añadido en educación*. Retrieved from <http://dialnet.unirioja.es/servlet/exttes?codigo=17011%5Cnhttp://dialnet.unirioja.es/servlet/tesis?codigo=17011&orden=0&info=link>
- Soria, J. ., & Vicente, E. (2002). Estudio de las aportaciones del Parque Natural de la Albufera de Valencia. *Limnetica*, 105–116.
- Soria, J. M. (2006). Past, present and future of la albufera of Valencia Natural Park. *Limnetica*, 25, 135–142.
- Team, R. D. C. (2000). Introducción a R. Notas sobre R: Un entorno de programación para Análisis de Datos y Gráficos. *Venables Copyright c R. Gentleman & R. Ihaka Copyright c, 1*, 100.
- Usaquén Perilla, O. L., Gómez, A. G., Gómez, A. G., Díaz, C. Á., & Cortezón, J. A. R. (2012). Methodology to assess sustainable management of water resources in coastal lagoons with agricultural uses: An application to the Albufera lagoon of Valencia (Eastern Spain). *Ecological Indicators*, 13(1), 129–143. <https://doi.org/10.1016/j.ecolind.2011.05.019>
- Vicente, E., & Miracle, M. (1992). The coastal lagoon Albufera de Valencia an ecosystem under stress. *Limnetica*, 8:87-100.
- Wetzel, R. . (2001). *Limnology: Lake & Rivers Ecosystems*. San Diego.
- White, H. (1990). Connectionist Nonparametric Regression: Multilayer Feedforward Networks. *Neural Networks*, 535–550.



ANEJOS

A. CÓDIGO REDES MULTILAYER PERCEPTRON

```
rm(list=ls())
library(plyr)
library(neuralnet)
library(TTR)
library(plot3D)
library(classInt)

setwd("C:/Users/lauraso/Desktop/CODIGO MLP")

#CARGAMOS LOS FICHEROS.csv

Data=as.data.frame(read.csv("Data.csv", head=TRUE, sep=";"))

#creamos las funciones de normalización y desnormalización, y NORMALIZAMOS LOS
DATOS PARA EVITAR ERRORES (Ux,Lx)

Ux= 1
Lx= 0

norm.fun= function(x){
  ((x*(Ux-Lx))+ (max(x)*Lx-min(x)*Ux))/(max(x)-min(x))
}

desnorm.fun1=function(x){
  ((x*(max(Data$Chla)-min(Data$Chla)))-(max(Data$Chla)*Lx-min(Data$Chla)*Ux))/(Ux-
Lx)
}

Data.norm= as.data.frame(apply(Data,2,norm.fun))

Train= Data.norm[1:1000,]
Test = Data.norm[1001:1918,]

#DISEÑO DE LA ESTRUCTURA DE LA RED Y ENTRENAMIENTO
#Rango del error cuadrático medio (MSE) en función del número de repeticiones, teniendo
la estructura fijada

p=10 #tamaño se la muestra
k=10 #nuemero de neuronas ocultas
m= 10 #número de repeticiones

w=ncol(Data)-1 #numero de inputs
error2= NULL
error3=NULL
BIC=NULL
box1=matrix(,nrow=m,ncol=k)
```

```

box2=matrix(,nrow=m,ncol=k)

#PRIMER BUCLE

for (s in 2:w){

  Chla= Train$Chla
  Train1= as.data.frame(cbind(Train[,2:s], Chla))
  Chla= Test$Chla
  Test1= as.data.frame(cbind(Test[,2:s], Chla))
  ii= names(Train1)
  y= 2/s

  if (y==1) {
    frml= Chla~ V1
  } else {
    frml = as.formula(paste("Chla ~", paste(ii[!ii %in% "Chla"], collapse = " + ")))
  }
  for (j in 1:k) {

    for(i in 1:m) {

      neuralmodel= neuralnet(frml,Train1, hidden=j, algorithm =
"rprop+",err.fct="sse",rep=i, stepmax = 2e9, act.fct= "logistic")
      nn=compute(neuralmodel,Test1[,1:(s-1)])
      nn= sapply(nn$net.result,desnorm.fun1)
      Chla= sapply(Test1$Chla,desnorm.fun1)
      MSE2= sum((Chla- nn)^2)/nrow(Test1)
      BIC1=nrow(Test1)*log(MSE2)+j*log(nrow(Test1))
      error2[i]=MSE2
      BIC[i]=BIC1
    }

    box1[,j] = error2
    box2[,j] =BIC
  }
write.table(box1, paste("Box1_", (s-1), ".txt" ))
write.csv2(box2, paste("Box2_",(s-1), ".txt"))

}

Escenario1= read.table("Box1_ 1 .txt", head=TRUE, sep = " ")
Escenario2= read.table("Box1_ 2 .txt", head=TRUE, sep = " ")
Escenario3= read.table("Box1_ 3 .txt", head=TRUE, sep = " ")
Escenario4= read.table("Box1_ 4 .txt", head=TRUE, sep = " ")
Escenario5= read.table("Box1_ 5 .txt", head=TRUE, sep = " ")
Escenario6= read.table("Box1_ 6 .txt", head=TRUE, sep = " ")
Escenario7= read.table("Box1_ 7 .txt", head=TRUE, sep = " ")
Escenario8= read.table("Box1_ 8 .txt", head=TRUE, sep = " ")

Escenario1=replace(Escenario1,Escenario1>100,NA)
Escenario2=replace(Escenario2,Escenario2>100,NA)

```



```
Escenario3=replace(Escenario3,Escenario3>100,NA)
Escenario4=replace(Escenario4,Escenario4>100,NA)
Escenario5=replace(Escenario5,Escenario5>100,NA)
Escenario6=replace(Escenario6,Escenario6>100,NA)
Escenario7=replace(Escenario7,Escenario7>100,NA)
Escenario8=replace(Escenario8,Escenario8>100,NA)

dimnames(Escenario1) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario2) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario3) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario4) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario5) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario6) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario7) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))
dimnames(Escenario8) = list(c("1","2","3","4","5","6","7","8","9","10")
, c("1","2","3","4","5","6","7","8","9","10"))

setwd("C:/Users/luraso/Desktop/CODIGO MLP/Graficos")

Ylim = range(Escenario1)
Xlim = 1:(ncol(Escenario1))
filename <- "BoxWhiskers_1"
filetype <- "png"
width.cm <- 14
height.cm <- 6
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis = 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)

boxplot(Escenario1,xlab='nº de neuronas ocultas',col='lightskyblue2',
border='black', ylab= "Coeficiente de correlación",
horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)

Ylim = range(Escenario2)
Xlim = 1:(ncol(Escenario2))
filename <- "BoxWhiskers_2"
filetype <- "png"
width.cm <- 14
height.cm <- 6
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis = 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)
```

```
boxplot(Escenario2,xlab='nº de neuronas ocultas',col='lightskyblue2',
        border='black', ylab= "Coeficiente de correlación",
        horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)
```

```
Ylim = range(Escenario3)
Xlim = 1:(ncol(Escenario3))
filename <- "BoxWhiskers_3"
filetype <- "png"
width.cm <- 14
height.cm <- 6
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)
```

```
boxplot(Escenario3,xlab='nº de neuronas ocultas',col='lightskyblue2',
        border='black', ylab= "Coeficiente de correlación",
        horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)
```

```
Ylim = range(Escenario4)
Xlim = 1:(ncol(Escenario4))
filename <- "BoxWhiskers_4"
filetype <- "png"
width.cm <- 14
height.cm <- 6
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)
```

```
boxplot(Escenario4,xlab='nº de neuronas ocultas',col='lightskyblue2',
        border='black', ylab= "Coeficiente de correlación",
        horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)
```

```
Ylim = range(Escenario5)
Xlim = 1:(ncol(Escenario5))
filename <- "BoxWhiskers_5"
filetype <- "png"
width.cm <- 14
height.cm <- 6
pointsize <- 8
```




```
x11(width = width.cm, height = height.cm, fontsize = fontsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)
```

```
boxplot(Escenario5,xlab='nº de neuronas ocultas',col='lightskyblue2',
border='black', ylab= "Coeficiente de correlación",
horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)
```

```
Ylim = range(Escenario6)
Xlim = 1:(ncol(Escenario6))
filename <- "BoxWhiskers_6"
filetype <- "png"
width.cm <- 14
height.cm <- 6
fontsize <- 8
x11(width = width.cm, height = height.cm, fontsize = fontsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)
```

```
boxplot(Escenario6,xlab='nº de neuronas ocultas',col='lightskyblue2',
border='black', ylab= "Coeficiente de correlación",
horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)
```

```
Ylim = range(Escenario7)
Xlim = 1:(ncol(Escenario7))
filename <- "BoxWhiskers_7"
filetype <- "png"
width.cm <- 14
height.cm <- 6
fontsize <- 8
x11(width = width.cm, height = height.cm, fontsize = fontsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)
```

```
boxplot(Escenario7,xlab='nº de neuronas ocultas',col='lightskyblue2',
border='black', ylab= "Coeficiente de correlación",
horizontal=FALSE)
axis(side = 1, at = 1:10, labels = c(1, 2, 3, 4, 5, 6, 7))
savePlot(filename=filename,type=filetype)
```

```
Ylim = range(Escenario8)
Xlim = 1:(ncol(Escenario8))
filename <- "BoxWhiskers_8"
filetype <- "png"
width.cm <- 14
height.cm <- 6
```

```

pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1, cex.main= 1, family='sans', mar = c(3,3,2,3), mgp =
c(2,0.5,0), tcl = - 0.3)

boxplot(Escenario8,xlab='nº de neuronas ocultas',col='lightskyblue2',
        border='black', ylab= "Coeficiente de correlación",
        horizontal=FALSE)
savePlot(filename=filename,type=filetype)

##MODELO RNA EN (t) MODELACIÓN CLOROFILA-A

s=4#nºinputs
Chla= Train$Chla
C_Finorg= Train$C_Finorg
C_O=Train$C_O
Train1= as.data.frame(cbind(Train[,2:3],C_O,C_Finorg, Chla))
Chla1= Test$Chla
C_O= Test$C_O
C_Finorg=Test$C_Finorg
Test1= as.data.frame(cbind(Test[,2:3],C_O,C_Finorg, Chla1))
j=4#numero de neuronas ocultas
frml=Chla ~ C_Norg+ C_Forg + C_O+ C_Finorg

neuralmodel= neuralnet(frml,Train1, hidden=j, algorithm = "rprop+",err.fct="sse",rep=10,
stepmax = 2e9, act.fct= "logistic")

filename <- "RNA, MLP"
filetype <- "png"
width.cm <- 8
height.cm <- 8
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1, cex.axis= 1, cex.lab=0.8, cex.main= 1, family='sans', mar = c(0.5,3.5,2,4), mgp =
c(3,1,0), tcl = - 0.3)

plot(neuralmodel, rep= "best")
savePlot(filename="ARN",type="png")

nn=compute(neuralmodel,Test1[,1:s])
nn1=compute(neuralmodel,Train1[,1:s])

nn= sapply(nn$net.result,desnorm.fun1)# test con una muestra con la cual la red no ha
trabajado
nn1= sapply(nn1$net.result,desnorm.fun1)#validación con la muestra de Train
Chla= sapply(Test1$Chla,desnorm.fun1)
Chla1= sapply(Train1$Chla,desnorm.fun1)

pr.errorrest=Chla-nn #Error Absoluto
pr.errortrain=Chla1-nn1 #Error Absoluto

```



```
setwd("C:/Users/lauraso/Desktop/CODIGO MLP/Graficos")

#EXPORTAR CSV CON LOS DATA.FRAME

m= do.call(rbind,neuralmodel$weights)
weights(neuralmodel, rep="best")
write.table(m,"Weights.txt", sep="\t")
write.table(nn, 'ResultsTest.txt', sep="\t")
write.table(nn1, 'ResultsTrain.txt', sep="\t")

## GRAFICOS

#1º ENTRENAMIENTO
Xlim <- Ylim <- range(1:250)

filename <- "R Humecal vs Simulados (ENTRENAMIENTO, MLP)"
filetype <- "png"
width.cm <- 4
height.cm <- 4
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1), mgp
= c(2,0.5,0), tcl = - 0.3)

x= data.frame(Chla1,nn1)
plot(x, xlab="Resultados Humecal", ylab="Simulados", xlim=Xlim, ylim=Ylim,
bty="n");par(new=T)
abline(a=0, b=1, col="grey", lty=1);par(new=F)
savePlot(filename=filename, type=filetype)

#2º VALIDACIÓN

Xlim <- Ylim <- range(1:250)

filename <- "R Humecal vs Simulados (VALIDACION, MLP)"
filetype <- "png"
width.cm <- 4
height.cm <- 4
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1), mgp
= c(2,0.5,0), tcl = - 0.3)

x= data.frame(Chla,nn)
plot(x, xlab="Resultados Humecal", ylab="Simulados", xlim=Xlim, ylim=Ylim,
bty="n");par(new=T)
abline(a=0, b=1, col="grey", lty=1);par(new=F)
savePlot(filename=filename, type=filetype)
```

REPRESENTACIÓN TEMPORAL DE LOS RESULTADOS Y EL ERROR ABSOLUTO

#1º ENTRENAMIENTO

```
Xlim <- c(1, nrow(Train1))
```

```
Ylim <- range(nn1)
```

```
filename2="Chl-a en la Albufera (ENTRENAMIENTO, MLP)"
```

```
filetype <- "png"
```

```
width.cm <- 4
```

```
height.cm <- 4
```

```
pointsize <- 8
```

```
x11(width = width.cm, height = height.cm, pointsize = pointsize)
```

```
par(cex = 1.3, cex.axis = 1.2, cex.lab = 1.2, cex.main = 1, family = 'sans', mar = c(3.5, 3.5, 2, 1), mgp = c(2, 0.5, 0), tcl = - 0.3)
```

```
end = nrow(Train1)
```

```
ts = ts(nn1, start = 1, end = end, frequency = 1, deltat = 1)
```

```
plot(ts, xlab = "", ylab = "Chl-a (µg/l)", xlim = Xlim, ylim = Ylim, type = "b", col = "blue")
```

```
lines(Chla1, col = "red", type = "p")
```

```
savePlot(filename = filename2, type = filetype)
```

```
Xlim <- Ylim <- range(cbind(pr.errortrain, pr.errortrain))
```

```
filename2="Error Absoluto (ENTRENAMIENTO, MLP)"
```

```
filetype <- "png"
```

```
width.cm <- 4
```

```
height.cm <- 2
```

```
pointsize <- 8
```

```
x11(width = width.cm, height = height.cm, pointsize = pointsize)
```

```
par(cex = 1.3, cex.axis = 1.2, cex.lab = 1.2, cex.main = 1, family = 'sans', mar = c(3.5, 3.5, 2, 1), mgp = c(2, 0.5, 0), tcl = - 0.3)
```

```
ts2 = ts(pr.errortrain, start = 1, end = end, frequency = 1, deltat = 1)
```

```
plot(ts2, xlab = "Patrones", ylab = "Error Absoluto", ylim = Ylim, type = "h", col = "orange")
```

```
lines(pr.errortrain, col = "black", type = "p")
```

```
abline(a = 0, b = 0, col = "grey", lty = 1)
```

```
savePlot(filename = filename2, type = filetype)
```

#2º VALIDACIÓN

```
Ylim = range(nn1)
```

```
Xlim <- c(1, nrow(Test1))
```

```
filename2="Chl-a en la Albufera (VALIDACIÓN, MLP)"
```

```
filetype <- "png"
```

```
width.cm <- 4
```

```
height.cm <- 4
```

```
pointsize <- 8
```

```
x11(width = width.cm, height = height.cm, pointsize = pointsize)
```

```
par(cex = 1.3, cex.axis = 1.2, cex.lab = 1.2, cex.main = 1, family = 'sans', mar = c(3.5, 3.5, 2, 1), mgp = c(2, 0.5, 0), tcl = - 0.3)
```

```
end = nrow(Test1)
```

```
ts = ts(nn, start = 1, end = end, frequency = 1, deltat = 1)
```

```
plot(ts, xlab = "", ylab = "Chl-a (µg/l)", ylim = Ylim, type = "b", col = "blue")
```



```
lines(Chla, col="red", type="p")
lty=c(1,1),lwd=c(2,2),col=c("red","blue"))
savePlot(filename=filename2, type=filetype)

Xlim <- Ylim <- range(cbind(pr.errorrest,pr.errorrest))

filename <- "Error Absoluto (VALIDACIÓN, MLP)"
filetype <- "png"
width.cm <- 4
height.cm <- 2
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1), mgp
= c(2,0.5,0), tcl = - 0.3)

ts2= ts(pr.errorrest,start=1, end=end,frequency=1, deltat = 1)
plot(ts2, xlab="Patrones", ylab="Error Absoluto",ylim= Ylim, type= "h", col="orange")
lines(pr.errorrest,col="black",type="p")
abline(a=0, b=0, col="grey", lty=1)
savePlot(filename=filename, type=filetype)

#####

#2º VALIDACIÓN
Ylim= range(nn2)
Xlim <- c(1, nrow(Datos_Con.norm))
filename2="Chl-a en la Albufera (VALIDACIÓN, MLP_OBS)"
filetype <- "png"
width.cm <- 4
height.cm <- 4
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1), mgp
= c(2,0.5,0), tcl = - 0.3)

end=nrow(Datos_Con.norm)
ts= ts(nn2,start=1, end=end,frequency=1, deltat = 1)
plot(ts, xlab="",ylab="Chl-a (µg/l)", ylim= Ylim, type= "b", col="blue")
lines(Chla2, col="red", type="p")
savePlot(filename=filename2, type=filetype)
```

B. CÓDIGO RED RESUCRRENTE JORDAN

```

###RECURRENTE Jordan (t-30),#####

rm(list=ls())
library(plyr)
library(neuralnet)
library(TTR)
library(plot3D)
library(RSNNS)

setwd("C:/Users/lauraso/Desktop/CODIGO (t-30)")

#CARGAMOS LOS FICHEROS.csv

Data=as.data.frame(read.csv("Data(t-30).csv", head=TRUE, sep=";"))

#creamos las funciones de normalización y desnormalización, y NORMALIZAMOS LOS
DATOS PARA EVITAR ERRORES (Ux,Lx)

Ux= 0.9
Lx= 0.1

norm.fun= function(x){
  ((x*(Ux-Lx)+(max(x)*Lx-min(x)*Ux))/(max(x)-min(x))
}

desnorm.fun1=function(x){
  ((x*(max(Data$Chla)-min(Data$Chla))-(max(Data$Chla)*Lx-min(Data$Chla)*Ux))/(Ux-
Lx)
}

Data.norm= as.data.frame(apply(Data,2,norm.fun))

Train= Data.norm[1:1000,]
Test = Data.norm[1001:1888,]

#DISEÑO DE LA ESTRUCTURA DE LA RED Y ENTRENAMIENTO

#Rango del error cuadrático medio (MSE) en función del número de repeticiones,
teniendo la estructura fijada

s=7 #nºinputs
Chla= Train$Chla
Train1=as.data.frame(cbind(Train[,2:s],Chla))

Chla= Test$Chla
Test1= as.data.frame(cbind(Test[,2:s], Chla))

```



```
j=8#numero de neuronas ocultas
frml=Chla ~ C_Norg + C_Forg +C_Ftotal+ C_O+ C_Finorg+ C_NH4

modelJordan=jordan(Train1[,1:(s-1)], Train1$Chla,
                    size=j, learnFuncParams=c(0.1), maxit=2e6,
                    inputsTest=Test1[,1:(s-1)],rep=10,
                    targetsTest=Test1$Chla, linOut=FALSE)

nn=apply(modelJordan$fittedTestValues,2,desnorm.fun1)

Chla=sapply(Chla,desnorm.fun1)
pr.errorrest=Chla-nn #Error Absoluto

setwd("C:/Users/lauraso/Desktop/CODIGO (t-30)/Red Recurrente")

#EXPORTAR CSV CON LOS DATA.FRAME

m= do.call(rbind,neuralmodel$weights)
weights(neuralmodel, rep="best")
write.table(m,"Weights.txt", sep="\t")
write.table(nn, 'ResultsTest.txt', sep="\t")

## GRAFICOS

#2º VALIDACIÓN

Xlim <- Ylim <- range(1:250)

filename <- "Observados vs Simulados (VALIDACION, [t+30])"
filetype <- "png"
width.cm <- 4
height.cm <- 4
pointsize <- 8
x11(width = width.cm, height = height.cm, pointsize = pointsize)
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1),
    mgp = c(2,0.5,0), tcl = - 0.3)

x= data.frame(Chla,nn)
plot(x, xlab="Resultados Humecal", ylab="Simulados", xlim=Xlim, ylim=Ylim,
     bty="n");par(new=T)
abline(a=0, b=1, col="grey", lty=1);par(new=F)
savePlot(filename=filename, type=filetype)
```

##REPRESENTACIÓN TEMPORAL DE LOS RESULTADOS Y EL ERROR ABSOLUTO

#2º VALIDACIÓN

```
Ylim= range(nn)
```

```
Xlim <- c(1, nrow(Test1))
```

```
filename2="Chl-a en la Albufera (VALIDACIÓN, [t+30])"
```

```
filetype <- "png"
```

```
width.cm <- 4
```

```
height.cm <- 4
```

```
pointsize <- 8
```

```
x11(width = width.cm, height = height.cm, pointsize = pointsize)
```

```
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1),
```

```
mgp = c(2,0.5,0), tcl = - 0.3)
```

```
end=nrow(Test1)
```

```
ts= ts(nn,start=1, end=end,frequency=1, deltat = 1)
```

```
plot(ts, xlab="",ylab="Chl-a ( $\mu\text{g/l}$ )", ylim= Ylim, type= "b", col="blue")
```

```
lines(Chla, col="red", type="p")
```

```
savePlot(filename=filename2, type=filetype)
```

```
Xlim <- Ylim <- range(cbind(pr.errorrest,pr.errorrest))
```

```
filename <- "Error Absoluto (VALIDACIÓN, [t+30])"
```

```
filetype <- "png"
```

```
width.cm <- 4
```

```
height.cm <- 2
```

```
pointsize <- 8
```

```
x11(width = width.cm, height = height.cm, pointsize = pointsize)
```

```
par(cex = 1.3, cex.axis= 1.2, cex.lab=1.2, cex.main= 1, family='sans', mar = c(3.5,3.5,2,1),
```

```
mgp = c(2,0.5,0), tcl = - 0.3)
```

```
ts2= ts(pr.errorrest,start=1, end=end,frequency=1, deltat = 1)
```

```
plot(ts2, xlab="Patrones", ylab="Error Absoluto",ylim= Ylim, type= "h", col="orange")
```

```
lines(pr.errorrest,col="black",type="p")
```

```
abline(a=0, b=0, col="grey", lty=1)
```

```
savePlot(filename=filename, type=filetype)
```