UNIVERSIDADE DA CORUÑA
Departamento de Matemáticas

Tesis Doctoral

Modelización estadística con Redes Neuronales. Aplicaciones a la Hidrología, Aerobiología y Modelización de Procesos

María Castellano Méndez

Diciembre 2009

UNIVERSIDADE DA CORUÑA
Departamento de Matemáticas

Tesis Doctoral

Modelización estadística con Redes Neuronales. Aplicaciones a la Hidrología, Aerobiología y Modelización de Procesos

Autora: María Castellano Méndez
Director: Wenceslao González Manteiga
Tutor UDC: Ricado Cao Abad

Diciembre 2009

UNIVERSIDADE DA CORUÑA
Departamento de Matemáticas

Tesis Doctoral

Modelización estadística con Redes Neuronales. Aplicaciones a la Hidrología, Aerobiología y Modelización de Procesos

Autora: María Castellano Méndez
Director: Wenceslao González Manteiga

Diciembre 2009

AGRADECIMIENTOS

En primer lugar deseo agradecer al director de esta memoria, D. Wenceslao González Manteiga, su dedicación y apoyo durante estos años.

Este trabajo ha sido financiado en parte por los proyectos del Ministerio de Ciencia e Innovación (MICINN), MTM2008-03010 y MTM2005-00820, así como el proyecto de la Comisión Interministerial de Ciencia y Tecnología (CICYT) ANACOM, CTQ2004-07811-C02-01.

Ha sido un largo viaje, como un tren de antaño de larga distancia. Un trayecto cuajado de paradas, donde han ido subiendo y bajando multitud de desconocidos destinados a escribir con tinta indeleble pequeños y grandes fragmentos de esta aventura.

Deseo agradecer su apoyo y amor incondicional a las dos personas que más quiero, mi pequeña gran familia, mi madre y mi hermano. Son el corazón y el alma de este trabajo, como lo son de cada cosa que he hecho y haré a lo largo de mi vida.

Tras la locomotora, quiero agradecer su ayuda a todos los demás; aquellos que me han tendido una mano en el viaje, brindándome su tiempo, sus ideas, y su compañía en días de trabajo y tardes de charlas; en lugares cercanos y lejanos, en despachos y cafeterías a lo largo del globo. Permitidme usar la excusa de mi legendaria falta de tiempo para no decir nombres; hemos sido cómplices en esto, y por supuesto yo nunca delataría a un compañero de felonías; vosotros ya sabéis quienes sois y lo agradecida que estoy por cada momento de complicidad compartido.

A todos, a los que estáis o ya os fuisteis, a los valientes que seguís en mi tren y a los que ya dejé atrás, muchas gracias por todo… Ha sido fantástico…

ÍNDICE.

Introducción

Motivación

Las redes neuronales constituyen una herramienta de análisis, modelización y predicción que se puede encontrar cómodamente integrada en muy diversos campos: robótica, ingeniería, psicología,… De este modo en sus aplicaciones a cada ámbito las redes adoptan connotaciones diferentes, y son vistas como herramientas de la ingeniería, réplicas del pensamiento racional, modelos de *caja negra*,… En todos los casos las redes se rigen por la filosofía general de obtener modelos coherentes con la realidad observada, de tal modo que sean los datos los que determinen el comportamiento de la red, bien a través de la determinación de sus estructura, bien de sus parámetros internos… Estas ideas acercan las redes neuronales a las ideas de los métodos *no paramétricos* de análisis de datos, en particular y en general al ámbito de la estadística.

Consideradas ya las redes neuronales como potentes herramientas estadísticas de análisis de datos, esta memoria se basa en dos motivaciones principales. En primer lugar se busca clarificar la naturaleza de las redes neuronales, analizando sus fundamentos, poniendo de manifiesto su naturaleza estadística y sus conexiones con diversos métodos estadísticos ampliamente conocidos. Resaltando sus principios estadísticos la visión de las redes neuronales se refuerza, ganando en consistencia teórica al tiempo que se mantiene su carácter intuitivo e innovador. En segundo lugar se desea mostrar con ejemplos concretos la bondad de las redes neuronales como herramientas estadísticas de modelización y su capacidad para responder a las necesidades que presentan los problemas observados en el mundo real. Con este fin se han seleccionado distintos problemas reales de ámbitos muy diversos, principalmente de la industria y el medioambiente y se han analizado y modelizado mediante redes neuronales.

Se puede apreciar, pues, que se trata de un trabajo con profunda vocación práctica, que busca no sólo realizar un estudio teórico de los distintos tipos de redes neuronales, y de sus conexiones con la estadística, desde la clásica a no paramétrica, sino también mostrar cómo las redes neuronales constituyen modelos capaces de dar el salto del ámbito teórico a la realidad, aportando soluciones a problemas reales, al tiempo que se mantiene su rigor y esencia.

Esquema de la Monografía

Esta tesis está estructurada en cuatro capítulos, además de esta breve introducción.
El primer capítulo responde al primero de los objetivos citados anteriormente, el conocimiento de los modelos de redes neuronales se centra por tanto en hacer una revisión de los orígenes de las redes neuronales, sus propiedades como modelos y sus relaciones con otros métodos estadísticos convencionales, más o menos avanzados. Este capítulo permite entender la filosofía de las redes neuronales, indagando en su naturaleza estadística, y proporcionándoles mayor profundidad que la que su vocación de "caja negra" les suele otorgar. Inicialmente se

presenta el origen de las redes neuronales, y las ideas latentes que subyacen en estos modelos; en la segunda sección se detalle el funcionamiento de las redes neuronales, se fijan las bases de la notación que se va a emplear en este documento, y se proporcionan ejemplos de los ámbitos donde su uso está extendido; la tercera sección proporciona una visión panorámica de la amplia variedad de modelos que responden al nombre de redes neuronales, a través de su organización y clasificación en diferentes categorías siguiendo múltiples criterios de organización; será en las secciones cuarta a sexta de este primer capítulo donde se analicen las conexiones que existen entre las redes neuronales y diferentes modelos estadísticos, de predicción y clasificación, paramétricos y no paramétricos, y se aportarán algunas variaciones de estos métodos que surgen de modo natural a partir del análisis de la estructura de las redes. Finalmente en la sección séptima de este primer capítulo se mostrará también una visión de las redes como aproximadores universales.

Los siguientes capítulos, del segundo al cuarto ilustrarán la capacidad de modelización de las redes neuronales a través de su aplicación a diversos problemas de diversa índole en ámbitos medioambientales e industriales; las aplicaciones comprenden problemas de regresión, problemas de clasificación y finalmente problemas de control de procesos. En éste último ámbito se estudiarán diversas posibilidades para la aplicación de la capacidad predictiva de los modelos de redes neuronales en problemas de control de procesos desde la perspectiva estadística y de ingeniería.

El segundo capítulo se centra en la aplicación de las redes neuronales a un problema de modelización en un proceso continuo. En este caso el problema se enmarca dentro al campo de la hidrología, en particular en el ámbito de la predicción de escorrentías o caudales de un río. El objetivo será comparar el funcionamiento de un modelo de redes neuronales con el de series de tiempo tradicionales, a la hora de enfrentarse a la predicción del caudal en la cuenca de un río; se trata de un problema real que tiene como fin proporcionar información vital para la gestión de conjunto de centrales hidroeléctricas, que se sitúan en un determinado cauce fluvial.

El tercer capítulo se centra en las aplicaciones binarias de las redes neuronales, abordando un problema de predicción de probabilidades futuras. En este caso el problema subyacente que motivó este problema se enmarcaba dentro del área de medioambiente y consistía en la predicción de la probabilidad de que la concentración de polen en el aire alcanzase ciertos niveles relevantes para la salud pública. Es por tanto un problema de clasificación entre días de riesgo alto, medio o bajo para los pacientes sensibles a la presencia de polen en el aire. Este ejemplo real se complementa con un estudio de simulación, en el que se compara el funcionamiento de una red con el de un Modelo Lineal Generalizado en el caso de respuesta binaria.

El cuarto capítulo está dedicado a la relación de las redes neuronales con las técnicas de control. En él se revisan las metodologías de control más interesantes al tiempo que aporta ideas de cómo introducir las redes neuronales en estas estructuras. Presenta dos ejemplos de aplicación a procesos industriales. Uno de ellos con respuesta de control discreta (colada en placa de cobre) y otro en un proceso de depuración de aguas. Como se señalará en el capítulo los trabajos realizados en el ámbito de la depuración de aguas no se limitaron al control ni a las

redes neuronales, si no que han abarcado técnicas de selección de variables, determinación de parámetros empleando técnicas bootstrap… aunque en esta tesis se presentarán solamente aquellos directamente relacionados con estos tópicos.

En cada uno de los capítulos se incluyen dos apartados finales, uno de resumen o conclusión de cada capítulo en el que se repasan las ideas más relevantes contenidas en el mismo, y finalmente la bibliografía, que se presenta de modo separado en cada capítulo, para facilitar su análisis.

CAPÍTULO 1.UNA REVISIÓN GENERAL

RESUMEN

Las Redes Neuronales Artificiales, RNA, constituyen una técnica de análisis de datos que desde hace algunos años se ha extendido con fuerza a los más diversos ámbitos. Este capítulo recoge una revisión de los modelos de redes neuronales más comunes. Así mismo mostrará el enfoque estadístico de las redes neuronales, permitiendo en muchas ocasiones su interpretación. Las redes neuronales se emplean en el reconocimiento de señales, en la simulación de sistemas biológicos, y en el análisis de datos. Esta última faceta nos las relaciona claramente con diversos métodos estadísticos no paramétricos tanto de regresión como de clasificación. Se presentarán diferentes redes neuronales que 'clonan' a los principales métodos no paramétricos, señalando los paralelismos entre los diferentes elementos que aparecen en los dos campos. Se estudiarán sus posibilidades como modelos generales de regresión y de clasificación.

1.1 Introducción

Desde los orígenes del hombre el deseo de observar, comprender, y cambiar su entorno para adaptarlo a sus necesidades ha sido el motor de su evolución. Desde las primeras herramientas de metal, las pieles curtidas, la rueda, el diseño de la máquina voladora de Leonardo, hasta la decodificación del genoma humano, el hombre analiza su entorno en busca de respuestas que mejoren la vida del ser humano, siempre tomando la naturaleza como modelo.

Aferrados a ese principio, a mediados del siglo XX surge un movimiento científico que trata de imitar una de las cualidades más fascinantes y al tiempo misteriosas del ser humano, su inteligencia; para ello se intenta construir máquinas cuya estructura funcional sea similar a la del cerebro humano, esperando que de esta idea surja tras un cierto período de aprendizaje, de modo natural la luz de la inteligencia; esta búsqueda se centra principalmente en la capacidad que tiene el ser humano para tomar decisiones de modo independiente.

El Reconocimiento de Muestras (Pattern Recognition) es la disciplina que responde al problema:

"Dando algunos ejemplos de señales complejas, y la correcta decisión para ellas, tomar de forma automática decisiones para una sucesión de futuros ejemplos."

El Reconocimiento de Patrones atañe a un extenso rango de actividades en muchos ámbitos de la vida. Algunos ejemplos serían:

- Graduar las distintas capas de una imagen visual
- Clasificación de tumores
- Nombrar distintas especies de plantas con flores

- Reconocimiento de escritura

Algunas de estas tareas forman parte de nuestra vida cotidiana. A través de nuestros sentidos recibimos datos, y a menudo somos capaces de identificar el origen de esos datos, de un modo inmediato y sin esfuerzo consciente (reconocer caras, voces,... incluso en malas condiciones). Como los humanos podemos hacer estas tareas, en ocasiones mejor que las máquinas, despertó un gran interés el conocer los principios que permiten que el ser humano realice estas tareas.

El cerebro humano tiene además otras características deseables, como su flexibilidad, la facilidad con que se adapta a nuevas situaciones (aprendiendo), la tolerancia a los fallos, la capacidad que tiene de manejar información "fantasma", esto es, no tangible, difusa,...

Durante años ingenieros, sicólogos y fisiólogos han intercambiado ideas sobre el funcionamiento de los cerebros de animales y hombres, con el fin de clarificar los complejos mecanismos que subyacen en la toma de decisiones y automatizarlos usando ordenadores. La primera máquina influenciada por las ideas de la biología y la psicología fue el PERCEPTRÓN (Rosenblatt, 1958), que despertó un gran interés en los años 60 debido a su capacidad para reconocer patrones sencillos. La segunda máquina fue creada, ya a mediado de los 80, con la herramienta de las redes neuronales. Ambas abandonaron rápidamente sus raíces biológicas para ser estudiadas desde un punto de vista matemático. El reconocimiento de patrones tiene una larga y respetable historia dentro de la ingeniería, especialmente en aplicaciones militares. El coste del hardware, tanto para adquirir los datos como para computar las respuestas, lo convirtió durante años en una área muy especializada. El avance del hardware hizo que aumentasen los intereses y las aplicaciones del Reconocimiento Muestral, revitalizando así su estudio.

Este estudio se centrará en las REDES NEURONALES ARTIFICIALES, (RNA), también denominadas ANN por su denominación en inglés, (Artificial Neural Networks); en concreto, en aquellas con conexiones hacia delante.

Se denominaron REDES NEURONALES a aquellos modelos nacidos con el fin de imitar el aprendizaje humano. Esta terminología se ha extendido a aquellos métodos que, en mayor o menor medida tienen como germen aquellas revolucionarias ideas.

El nombre Redes Neuronales surge de la analogía con el cerebro humano y el modo en que los hombres pueden acercarse a la tarea de reconocer muestras. Un largo camino las separa ya de sus raíces biológicas. Se han hecho grandes avances, y aunque muchos de ellos se han opuesto a un cuidadoso escrutinio, los métodos de las Redes Neuronales han tenido un gran impacto en la práctica del reconocimiento de patrones (Haykin, 2009; Tang *et al*, 2007). Las redes neuronales artificiales pueden considerarse como una herramienta estadística completa para el análisis de datos (Bishop, 1995)

El entendimiento teórico de cómo trabajan está todavía en proceso de construcción y abordaremos aquí su estudio desde un enfoque estadístico. La filosofía de las redes se basa en que sean los datos los que establezcan el comportamiento de la red a través de un aprendizaje, y evitar así estar sujetos a una estructura encorsetada. Estas ideas acercan las redes neuronales a la filosofía de los métodos *no paramétricos* de análisis de datos. Estos métodos tienen la característica de poder aproximar funciones de muy diversa índole, soslayando la necesidad de establecer un modelo rígido al que ajustar los datos.

En la segunda sección se explicará con detalle en qué consisten las redes neuronales, también llamadas artificial neural networks (RNA), se fijará la notación que emplearemos al exponer los distintos tipos de redes, y señalaremos los principales ámbitos en los que se han venidos utilizando las redes neuronales.

En la tercera sección realizaremos una minuciosa clasificación de las redes neuronales según diferentes criterios, como su topología, el tipo de conexiones que presenta la red, y el tipo de aprendizaje que se emplee en el entrenamiento. Este último enfoque nos proporcionará a su vez diversas clasificaciones derivadas de la existencia de diferentes criterios de clasificación del aprendizaje.

Las secciones cuarta y quinta podrán de manifiesto los sólidos vínculos existentes entre las redes neuronales y las reglas de predicción y clasificación. Se expondrán las redes asociadas a muchos y muy diversos métodos estadísticos de clasificación y predicción, con el fin de ilustrar adecuadamente cómo las redes neuronales pueden representar gran parte de las técnicas estadísticas. En particular se estudiarán las relaciones entre las redes neuronales de base radial y diferentes métodos no paramétricos tanto de regresión como de estimación de la función de la densidad, mostrando algunas variaciones de estos métodos surgidas de las peculiaridades inherentes a las redes neuronales. La sección sexta muestra algunos ejemplos de estadística multivariante concretos, mientras en la sección séptima se mostrará también una visión de las redes como aproximadores universales, que se ilustrará con algunos ejemplos. A continuación la octava sección revisa otra de las estructuras de aprendizaje más extendidas, las llamadas redes probabilísticas, para terminar con una sección resumen de lo expuesto. Este primer capítulo constituye la base necesaria para la construcción de aplicaciones de las redes neuronales a problemas reales, lo que constituirá el cuerpo de la presente monografía.

1.2. ¿Qué es una red neuronal?

Una RED NEURONAL es un proceso sólido y paralelamente distribuido con la propensión natural a acumular procedimientos experimentales y hacerlos disponibles para su uso. Se parece al cerebro en dos aspectos, por una parte la red adquiere conocimientos a través de un proceso de aprendizaje, y por otra las conexiones interneuronales, conocidas como *cargas sinápticas* presentan una gran solidez de se encargan de almacenar los conocimientos.

El funcionamiento de una red sería el siguiente. Se dispone de una serie de datos (situaciones del pasado) y asociados a ellos la respuesta deseable de la red (*training set*). La red de algún modo observa estos hechos y aprende de ellos (entrenamiento o *aprendizaje*), de modo que cuando se encuentre en una nueva situación actúe de modo coherente con lo aprendido. Para evaluar el comportamiento de la red ante nuevas situaciones se considerará un nuevo subconjunto de datos (*validation set*), independiente del conjunto de entrenamiento.

1.2.1 Terminología y Notación

Del mismo modo que el cerebro está constituido por neuronas, la unidad básica constituyente de la red neuronal es el nodo, (neurona o elemento de procesado) Un nodo es un elemento de cálculo interconectado con otros muchos elementos, imitando las sinapsis nerviosas. La idea era que, tal vez, conectando un número suficientemente alto de neuronas o nodos la

inteligencia naciese de modo natural del aprendizaje. A un nodo llegan conexiones desde muchas otras neuronas, y en general proporciona una única salida, como muestra la Figura 1.1.

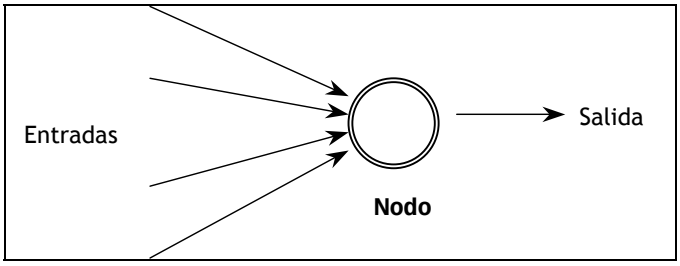

Figura 1.1. Esquema de un Nodo

A un nodo pueden llegar muchas entradas, x_i, proporcionando gran cantidad de información. El nodo condensará esta información de algún modo, por ejemplo a través de una combinación lineal con cierto sesgo, (entrada ponderada); a continuación se aplica cierta función, obteniéndose de este modo la salida del nodo, que a su vez podrá ser la entrada de algún otro nodo.

$$I = \omega_0 + \sum_{i=1}^{N} \omega_i x_i \tag{1.1}$$

$$f(I) = f\left(\omega_0 + \sum_{i=1}^{N} \omega_i x_i \right) \tag{1.2}$$

Los parámetros ω_i reciben el nombre de *pesos*, y reflejan la conexión entre las neuronas y cuán fuerte es. Por su parte la función *f* recibe el nombre de función de transferencia o link; la adecuada elección de esta función es en parte responsable de la bondad del comportamiento de la red neuronal.

Algunas de las funciones de transferencia más usuales son:

Umbral: $f(x) = \begin{cases} 0 & si \quad x < 0 \\ 1 & si \quad x \geq 0 \end{cases}$ (1.3)

Lineal o Identidad: $f(x) = x$ (1.4)

Tangente Hiperbólica: $f(x) = Th(x) = \dfrac{e^{2x} - 1}{e^{2x} + 1}$ (1.5)

Logística: $f(x) = \dfrac{1}{1 + e^{-x}} = \dfrac{1 + Th(x/2)}{2}$ (1.6)

Gaussiana: $f(x) = e^{\frac{-x^2}{2}}$ (1.7)

Algunas de estas funciones tienen la particularidad de que llevan al conjunto de entradas a un espacio acotado, donde las funciones presentan cualidades que pueden resultarnos de interés en determinados contextos. Cada nodo puede tener una función de transferencia propia,

distinta a la de los demás nodos, aunque en la mayoría de los casos los nodos situados de un mismo nivel presentan todos la misma función de transferencia.

En las redes neuronales con conexiones hacia adelante los nodos suelen estar distribuidos en distintas capas, de modo que los de una capa están conectados sólo con los de la capa inmediatamente superior. Así la capa de entrada es aquella en que los datos, se presentan a la red; las variables que conforman estos datos de entrada reciben el nombre de *inputs*; por otra parte, la capa de salida es aquella en que la red nos devuelve su respuesta o respuestas, que reciben el nombre de *outputs*.

Finalmente las capas intermedias reciben el nombre de capas ocultas. En general fijamos la restricción de que desde la capa de salida no puede surgir ninguna conexión a otro nodo, para evitar bucles continuos en la red

Establezcamos la **notación** que vamos a manejar en el estudio de las redes neuronales.

(a) Dimensiones de la red y notación de las variables

- $\boldsymbol{L_H}$ es el número de capas ocultas
- $\boldsymbol{N_I}$ es el número de variables explicativas o entradas (*inputs*). Coincide con el número de nodos de la capa de entrada
- $\boldsymbol{N_{H^l}}$ es el número de nodos de la *l*-ésima capa oculta, $1 \le l \le L_H$
- $\boldsymbol{N_O}$ es el número de nodos de la capa de salida
- $\boldsymbol{X_i}$ son las variables explicativas, $1 \le i \le N_I$
- $\boldsymbol{Y_k}$ son las variables objetivo, $1 \le k \le N_O$

(b) Pesos de la red

- $\boldsymbol{\omega_{0j}^{h_{l-1}h_l}}$ sesgo de entrada al nodo j-ésimo de la l-ésima capa oculta $1 \le l \le L_H \quad 1 \le j \le N_{H^l}$
- $\boldsymbol{\omega_{ij}^{ah_1}}$ pesos desde la capa de entrada a la l-ésima capa oculta $1 \le i \le N_I \quad 1 \le j \le N_{H^l}$ con $1 \le l \le L_H$
- $\boldsymbol{\omega_{ij}^{h_{l_1}h_{l_2}}}$ pesos desde la capa l_1-ésima capa oculta hasta la l_2-ésima capa oculta, $\boldsymbol{1 \le i \le N_{H^{l_1}}}$, $\boldsymbol{1 \le j \le N_{H^{l_2}}}$ con $\boldsymbol{1 \le l_1 \le L_H}$ y $\boldsymbol{1 \le l_2 \le L_H}$
- $\boldsymbol{\omega_{0k}^{h_{L_H}o}}$ sesgo de entrada al nodo k-ésimo de la capa de salida $\boldsymbol{1 \le k \le N_O}$
- $\boldsymbol{\omega_{ik}^{ao}}$ pesos desde la capa de entrada a la capa de salida $\boldsymbol{1 \le i \le N_I}$ y $\boldsymbol{1 \le k \le N_O}$

- $\omega_{jk}^{h,o}$ pesos desde la l-ésima capa oculta al nodo k-ésimo de la capa de salida, $1 \le j \le N_{H^l}$ y $1 \le k \le N_O$ con $1 \le l \le L_H$

(c) Valores de la red y funciones de transferencia:

- g_j^l es la entrada ponderada al nodo j-ésimo de la l-ésima capa oculta $1 \le j \le N_{H^l}$ y $1 \le l \le L_H$
- h_j^l es la salida del nodo j-ésimo de la l-ésima capa oculta $1 \le j \le N_{H^l}$ y $1 \le l \le L_H$
- q_k es la entrada al nodo k-ésimo de la capa de salida $1 \le k \le N_o$
- O_k es la salida o predicción k-ésima de la capa de salida $1 \le k \le N_o$
- r_k es el residuo o error k-ésimo; $1 \le k \le N_o$, $r_k = \left(Y_k - O_k\right)$
- $f_{h^l}^j$ es la función link asociada al nodo j-ésimo de la l-ésima capa oculta $1 \le j \le N_{H^l}$ y $1 \le l \le L_H$
- f_o^k es la función link asociada al nodo k-ésimo de la capa de salida; $1 \le k \le N_O$
- f_{h^l} es la función link asociada a la l-ésima capa oculta; $1 \le l \le L_H$
- f_o es la función link asociada a la capa de salida

La mayor parte de los modelos de redes neuronales, pueden ser expuestos como diagramas de red, lo que facilita la comprensión y el análisis de su estructura. A continuación se expone cómo se representan gráficamente los distintos elementos que conforman la red

(d) Neuronas. Cada neuronas o nodo se representa mediante círculos y/o cajas.

- **CÍRCULOS**: son variables observadas o salidas de nodos, identificadas con su nombre.

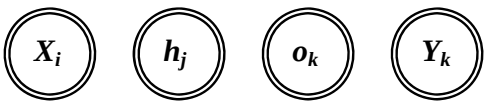

Figura 1.2. Representación Circular de Nodos

- **CAJAS:** son otro modo de representar los valores calculados como función de uno o más argumentos. Dentro tendrán un símbolo indicador del tipo de función de transferencia empleada. Las cajas tienen además un parámetro asociado llamado sesgo.

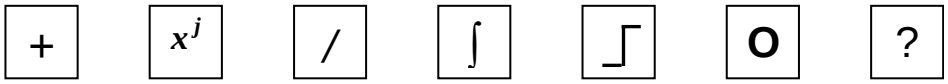

Figura 1.3. Representación de las Operaciones

Las operaciones representadas en las Figura 1.3 son, de izquierda a derecha, suma, potencia, combinación lineal, función logística, función umbral, función de base radial y un valor arbitrario.

(e) **CONEXIONES y AJUSTES. Las conexiones entre neuronas se representan por flechas.**

- **FLECHAS:** indican que el origen de la flecha es un argumento de la función que se calcula en el destino de la flecha.
- **DOS LÍNEAS PARALELAS:** indican que los valores de cada final han de ser ajustados, por mínimos cuadrados, máxima verosimilitud, o algún otro criterio de estimación.

1.2.2 Los usos de las Redes Neuronales

Las RNA se usan principalmente con tres objetivos diferenciados:

- *Como modelos nerviosos biológicos, e " inteligencia".* Parte de las motivaciones biológicas que dieron lugar a las redes neuronales se han conservado, por ello y se siguen empleando como instrumentos que nos ayuden a entender y duplicar el funcionamiento de los sistemas nerviosos de los seres vivos.
- *Como procesadores adaptativos de señal en tiempo real, o controladores, implementados en hardware para aplicaciones como robots. Esta es el área del Reconocimiento de Patrones. La tecnología, la ciencia y los negocios han aportado nuevas tareas de interés (diagnóstico de enfermedades, leer códigos ZIP...), que en algunos casos son eminentemente tecnológicas, como la lectura de códigos de barras,* y en otros muchos las llevan a cabo expertos humanos. El objetivo es construir máquinas que realicen estas labores de un modo más "*rápido*", más "*barato*", y más "exacto" que los hombres. Cada vez es más factible idear sistemas automáticos que sustituyan y mejoren al especialista (como la "cuenta de crédito de un cliente"), o clonar al experto (ayuda al diagnóstico médico)
- *Como métodos de análisis de datos.* Principalmente este trabajo se centrará en esta última faceta de las redes. De hecho en multitud de ocasiones si se analiza detalladamente lo que está haciendo una red se descubrirá que se dedica a rescribir métodos estadísticos clásicos, como se detallará en secciones posteriores. Así mismo cualquier red, aún cuando su objetivo final sea algo muy concreto y a simple vista alejado de la estadística, como "la voz" del desfibrilador que nos dice que el paciente tiene pulso y no puede recibir la descarga, fue diseñada, y por lo tanto puede ser analizada desde un punto de vista meramente estadístico. Esta faceta de las redes fue estudiada con detenimiento por Sarle (1994).

Se continua buscando el modelo que nos permita crear inteligencia, en el sentido humano del término, la llamada inteligencia artificial; una máquina capaz de aprender realmente, de tomar sus propias decisiones, de modificar y reinventar sus reglas de aprendizaje,...

1.3. Clasificación de las Redes Neuronales

La principal clasificación de las redes se basa en las conexiones que presentan. En esta tesis se trabajara con redes que presentan conexiones "*hacia adelante"* o "*feedforward*". Estas redes así mismo pueden ser clasificadas según dos criterios principales: la arquitectura o topología de la red, y el método de aprendizaje empleado para su entrenamiento.

1.3.1 Según sus Conexiones

Un criterio fundamental para clasificar las redes es aquel que se basa en las conexiones o nodos que presentan. A pesar de que en la mayoría de los casos se tratará con redes con conexiones "*hacia delante*", (*feedforward*), en particular aquellas que solo presentan conexiones entre capas consecutivas, existen redes que no limitan de esa manera sus conexiones.

1.3.1.1 Redes con conexiones "*hacia adelante*" (feedforward)

En este tipo de redes la información se propaga hacia adelante por las distintas capas a través de los pesos. No existen conexiones "*hacia atrás*", ni "*laterales*" (salvo en dos casos particulares propuestos por Kohonen, que presentan conexiones implícitas entre las salidas, que son el Learning Vector Quantizer (LVQ), y el Topology Preserving Map (TRM)) (Hilera González y Martínez Hernando, 1995). Redes de este tipo serán las que se consideren de ahora en adelante.

1.3.1.2 Redes con conexiones "*hacia adelante*" y "*hacia atrás*" (feedforward/feedback)

Son redes donde la información circula tanto hacia adelante como hacia atrás, pues existen conexiones, *i.e.* pesos, en ambos sentidos. Entre dos neuronas conectadas hay dos pesos, uno en cada sentido, que en la mayoría de los casos son diferentes. Generalmente son redes bicapa. Muchas de estas redes basan su comportamiento en la resonancia, esto es, en la interacción de las informaciones de la primera y la segunda capa, hasta alcanzar un estado estable.

En ocasiones se dan conexiones laterales entre neuronas de una misma capa. A este tipo de redes pertenecen las red Adaptative Resonance Theory (ART), y la red Bidirectional Associative Memory (BAM) (Hilera González y Martínez Hernando, 1995).

1.3.2 Según su Topología

Las dos arquitecturas de redes neuronales más usada son los Perceptrones Multicapa, denominados habitualmente MLP debido a las siglas de su denominación anglosajona, "*Multilayer Perceptron*", y las Funciones de Base Radial, que se asocian a sus siglas en inglés, "*Radial Basis Functions Básicas*".

1.3.2.1 El Perceptrón

El primer modelo de red que se diseñó fue el perceptrón. Fue inventado por F. Rosenblatt en 1958. Con él pretendía ilustrar algunas de las propiedades de los sistemas inteligentes. Posteriormente se desarrollaron además diversas variantes del perceptrón, en particular el perceptrón simple, esto es, sin capa oculta, entrenado según una regla delta (con supervisión) La gran flexibilidad de este primer esquema influyó enormemente en el gran desarrollo posterior de lo que acabó desembocando en las redes neuronales. Los perceptrones se clasifican por el número de capas que presentan. Así aquellos con dos capas, esto es, sin capa oculta, son perceptrones simples, y los de una o más capas ocultas se llaman perceptrones multicapa.

A continuación se ilustra el ejemplo más sencillo: el perceptrón simple. Un perceptrón simple calcula la combinación lineal de las entradas (con un término de sesgo) lo que se llama *entrada de red* (1.8); a esa combinación lineal se aplica una función de activación, por regla general la función signo, o la función umbral, dando lugar a la salida de la red. La figura 1.4 muestra el diseño de un perceptrón simple.

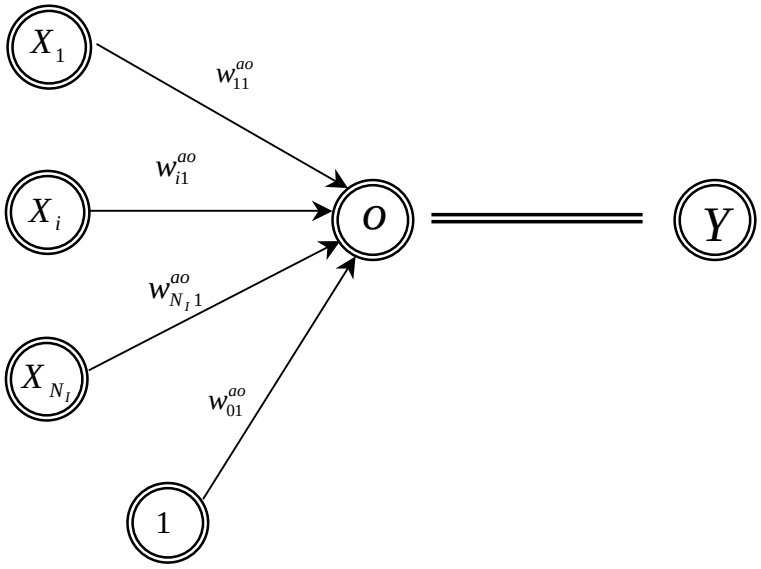

Figura 1.4. Esquema del Perceptron Simple

Las ecuaciones del proceso son las siguientes

$$\boldsymbol{q = \omega_{01}^{ao} + \sum_{i=1}^{N_I} \omega_{i1}^{ao} X_i} \tag{1.8}$$

$$O = f(q) = \begin{cases} 0 & si \quad q < 0 \\ 1 & si \quad q \geq 0 \end{cases} = \begin{cases} 0 & si \quad \omega_{01}^{ao} + \sum_{i=1}^{N_I} \omega_{i1}^{ao} X_i < 0 \\ 1 & si \quad \omega_{01}^{ao} + \sum_{i=1}^{N_I} \omega_{i1}^{ao} X_i \geq 0 \end{cases} \tag{1.9}$$

Está constituido por N_I nodos de entrada y una única neurona de salida, encargada de decidir a cuál de las dos clases posibles pertenece la observación.

La regla de decisión será *1* si la observación pertenece a la clase *A*, y *0* si pertenece a la clase *B*. La salida dependerá de los pesos ω_{i1}^{ao} y del sesgo ω_{01}^{ao}, que en este caso cumple el papel de valor umbral. Para que el clasificador pueda clasificar correctamente cualquier muestra es necesario que las dos clases sean linealmente separables.

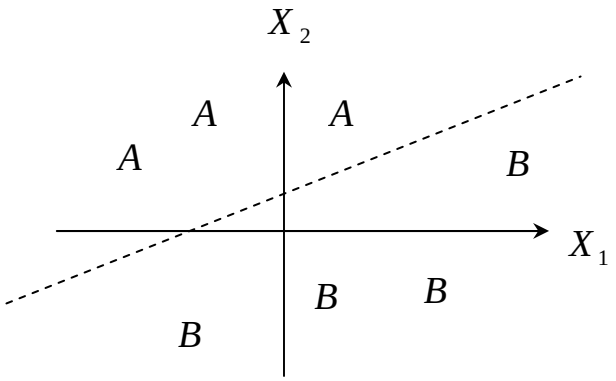

Figura 1.5. Separación lineal en el plano de dos conjuntos

Las redes de este tipo, con sólo dos capas se limitan a la resolución de problemas con observaciones separables geométricamente (por hiperplanos). Ante estos inconvenientes surgieron dos modelos nuevos, el ADALINE (elemento lineal adaptable) y el MADALINE (elemento lineal adaptable múltiple). La estructura del ADALINE es la del perceptrón simple, pero la función que aplica el nodo es la identidad; de este modo se permite más flexibilidad en la estructura de la red.

Un modelo más versátil y complejo es el Perceptrón Multicapa (MLP), que consiste en cierta cantidad de nodos organizados por capas (en al menos 3 capas), de modo que una neurona reciba entradas sólo de las neuronas situadas en la capa inmediatamente inferior. En general, en un Perceptrón Multicapa cada uno de los nodos calcula una combinación lineal de las entradas que llegan a él, le añade un sesgo, y finalmente le aplica una función de activación, también llamada de transferencia, que por regla general traslada cualquier entrada real a un rango generalmente acotado, dando lugar así a la salida del nodo, que puede ser una de las entradas de un nuevo nodo.

Un perceptrón multicapa, al igual que uno simple puede tener una o más salidas, cada una de ellas con un conjunto de pesos y un sesgo asociados. A menudo se usa la misma función de activación para cada nodo de la misma capa, aunque es posible usar diferentes funciones de activación para cada neurona.

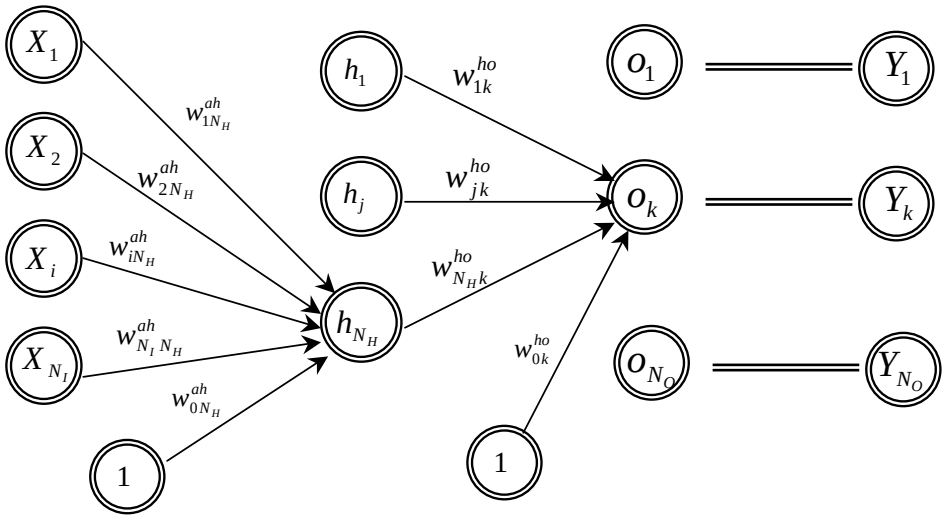

Figura 1.6. Esquema de un Perceptron Multicapa, con una capa oculta (N_I-N_H-N_O)

Las ecuaciones asociadas a un perceptrón multicapa son las siguientes:

$$h_j = f_h\left(\sum_{i=1}^{N_I} \omega_{ij}^{ah} \cdot X_i + \omega_{0j}^{ah}\right) \quad \text{para } j = 1,\cdots,N_H \tag{1.10}$$

$$o_k = f_O\left(\sum_{j=1}^{N_H} \omega_{jk}^{ho} \cdot h_j + \omega_{jk}^{ho}\right) \quad \text{para } k = 1,\cdots,N_O \tag{1.11}$$

En esta notación se supone que todos los nodos de una misma capa emplean la misma función de activación, aunque podría perfectamente no ser así.

1.3.2.2 Redes Neuronales de Base Radial

El otro modelo arquitectónico importante es el de las redes de base radial (RBF). La filosofía general de las redes consiste en huir de los modelos preestablecidos, y dejar que sean las observaciones pasadas las que el comportamiento de las salidas de la red. En los perceptrones esa influencia radica en el entrenamiento; en estas nuevas redes también, pero además se desean establecer ciertos valores de las variables de entrada y sus correspondientes variables respuesta de tal forma que sean representativos de todos los estados en los que se puede encontrar el sistema que se desea modelizar. Lo que va a diferenciar a estas redes de los perceptrones es el modo en que actúan sobre los datos de entrada, esto es, cómo condensan la información que les proporcionan las distintas variables. En un MLP la entrada de red (net input) a la capa oculta es una combinación lineal de las entradas especificada por los pesos. En una red de función de base radial las neuronas de la capa oculta calculan las funciones radiales básicas de las entradas, que son similares a las funciones empleadas en la regresión tipo núcleo (Härdle, 1990). Para ello será necesario disponer de un conjunto de observaciones, tal y como se tiene en la regresión no paramétrica, con respecto a los que calculamos la distancia del vector de entradas.

Ese conjunto de *centros* $\{\vec{W}_i\}_{i=1}^{N_H} = \{\vec{W}_i^{ah}, \vec{W}_i^{ho}\}_{i=1}^{N_H}$, siendo $\{\vec{W}_i^{ah}\}_{i=1}^{N_H} = \{(\omega_{1i}^{ah}, \omega_{2i}^{ah}, \ldots, \omega_{N_I i}^{ah})\}_{i=1}^{N_H}$, y $\{\vec{W}_i^{ho}\}_{i=1}^{N_H} = \{(\omega_{i1}^{ho}, \omega_{i2}^{ho}, \ldots, \omega_{iN_O}^{ho})\}_{i=1}^{N_H}$), tiene que cumplir una de las propiedades principales de los conjuntos de entrenamiento, ser significativos, esto es, que representen todas las situaciones en las que se puede encontrar el sistema que se desea imitar. Pero al contrario que en el caso de conjunto de entrenamiento deseamos reducir al máximo el número de elementos de ese conjunto, pues el número de pesos involucrados en la red será proporcional al número de centros escogidos. Además ha de ser independiente del conjunto de entrenamiento y del de validación.

Cuando se introduce un caso nuevo en la red, se calculan las distancias a los centros, que se matizarán en función de unos parámetros llamados "*ventanas*", ω_{0j}^{ah} , asociados a cada nodo oculto, y que cumplen tareas similares a las que cumple el parámetro "*ventana*" en la metodología tipo núcleo (Wand y Jones, 1995). La función de activación de la capa oculta (igual para todos los nodos) puede ser cualquiera de una variedad de funciones en los reales, que alcanzan el máximo en el 0, y que a medida que se acercan a $\pm\infty$, tienden a cero.

Por su parte en la capa de salida se calcularán combinaciones lineales de las salidas de la capa oculta, esto es, la función de activación será la identidad. En ocasiones se considera como ventana, la mitad del parámetro, pues es la amplitud de la zona a cada lado, en cualquier caso ambas ideas son equivalentes. Algunas posibilidades para las funciones K son:

Gaussiana $$K(r) = \exp\left(r^2/2\right) \tag{1.12}$$

Multicuadrática $$K(r) = \sqrt{(c^2 + r^2)} \tag{1.13}$$

Thin Plate Spline $$K(r) = r^2 \log r \tag{1.14}$$

En general se puede aplicar cualquier función tipo núcleo, pues se rigen por el mismo principio: establecer regiones en el espacio de entradas, que pueden superponerse unas a otras, entorno a ciertos puntos (centros) que se suponen significativos. La norma empleada para calcular la distancia entre un punto y los centroides no es fija, sino que constituye otro grado de libertad de la red. Las más utilizadas son la Euclídea y la de Mahalanobis. La Figura 1.7 muestra la estructura de una red RBF, y las ecuaciones siguientes muestran un ejemplo de la forma numérica que adopta.

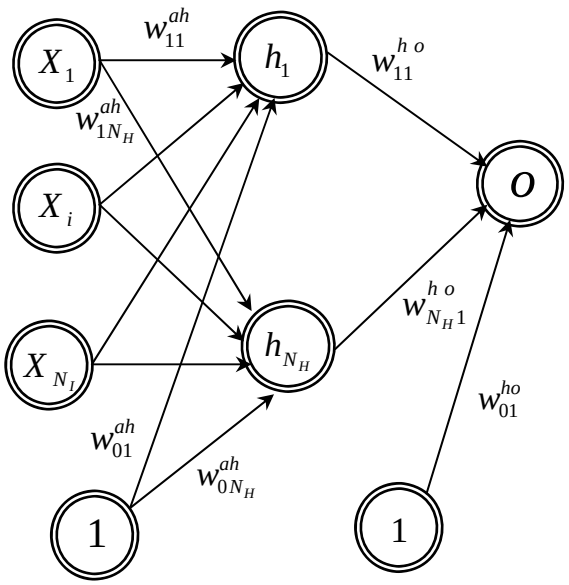

Figura 1.7. Esquema de una Red de Base Radial Múltiple con Salida Unidimensional

Siendo $\{X_i\}_{i=1}^{N_I}$ el conjunto de variables de entrada, y sea $\{\vec{W}_i\}_{i=1}^{N_H} = \{\vec{W}_i^{ah}, \vec{W}_i^{ho}\}_{i=1}^{N_H}$ el conjunto de los centros, se tiene,

$$h_i = K\left(\frac{1}{\omega_{0j}^{ah}}\left(\sum_{i=1}^{N_I}\left(\omega_{ij}^{ah} - x_i\right)^2\right)^{1/2}\right) \quad para\ i = 1,\ldots,N_H\ , \tag{1.15}$$

con N_H el número de nodos de la capa oculta, o lo que es lo mismo el número de centros que se ha establecido. La salida responde a la expresión:

$$o = \omega_{01}^{ho} + \sum_{i=1}^{N_H}\omega_{i1}^{ho}h_i = \omega_{01}^{ho} + \sum_{i=1}^{N_H}\omega_{i1}^{ho}K\left(\frac{1}{\omega_{0j}^{ah}}\left(\sum_{i=1}^{N_I}\left(\omega_{ij}^{ah} - x_i\right)^2\right)^{1/2}\right) =$$
$$= \omega_{01}^{ho} + \sum_{i=1}^{N_H}\omega_{i1}^{ho}K\left(\frac{\left\|\vec{x} - \vec{W}_i^{ah}\right\|}{\omega_{0i}^{ah}}\right) \tag{1.16}$$

La región cerca de cada centro de los RBF recibe el nombre de *campo receptivo* (receptive field) de la neurona oculta. Es la zona donde ejerce su influencia el centro asociado a ese nodo. Las neuronas RBF también se llaman *campos receptivos localizados* ("*locally tunned processing units*" o funciones de potencial) (Buhmann, 2003). En ocasiones los valores de la capa oculta se normalizan para que sumen 1, como se hace comúnmente en estimación tipo núcleo (Silverman, 1986).

Las redes RBF poseen la particularidad de que el número de nodos de la capa oculta coincide con el número de centros por lo que es imprescindible haber seleccionado el número de nodos

para tener la topología definitiva de la red. Es necesario pues abordar el problema de la elección de los centros. En principio se pueden considerar todos los pesos como susceptibles de ser modificados durante el entrenamiento, pero esta no es la única posibilidad.

Frecuentemente las redes RBF son consideradas como híbridas. Antes de comenzar el entrenamiento se realiza un análisis cluster (Everitt *et al.*, 2001; Peña, 2002) sobre el conjunto de entradas y se seleccionan como centros las medias de los cluster. Siguiendo esta misma idea las ventanas se toman a menudo de tal forma que coincidan con la distancia al k-ésimo vecino más cercano desde el centro seleccionado previamente, o bien se determinan a partir de las varianzas de los clusters. De este modo ni los centros ni las ventanas se determinan convenientemente de modo previo a la red. Durante el entrenamiento se buscan únicamente los valores de los parámetros que unen la capa oculta con la capa de salida, esto es, si se tratase de un problema de regresión, los valores que toma la variable objetivo en los centros.

Un tipo especial de redes de base radial son aquellas denominadas "*funciones potencial*". Estas funciones constituyen métodos tipo núcleo, de modo que cada observación o centro se considera asociado a una carga de intensidad, q_i. El potencial de un nuevo punto responde a la expresión (1.17), en la que *K* es una función tipo núcleo, y el potencial es seleccionado según los objetivos de la red.

$$f\left(\vec{X}\right)=\sum_{i=1}^{N_H} q_i \cdot K\left(\vec{X};\vec{W}_i^{ah}\right) \tag{1.17}$$

Las diferencias entre ambas topologías radica como en el modo de procesar la información de los nodos de la capa oculta de ambos, que se refleja en sus expresiones matemáticas.

MLP: $$\boldsymbol{g_j = \omega_{0j}^{ah} + \sum_{i=1}^{N_I} \omega_{ij}^{ah} x_i} \tag{1.18}$$

$$\boldsymbol{h_j = f\left(g_j\right)} \tag{1.19}$$

RBF: $$g_j = \left(\sum_{i=1}^{N_I}\left(\frac{\omega_{ij}^{ah}-x_i}{\omega_{0j}^{ah}}\right)^2\right)^{1/2} = \left\|\vec{W}_j^{ah}-\vec{X}\right\| \tag{1.20}$$

$$\boldsymbol{f_j = K\left(g_j\right)} \tag{1.21}$$

Tanto los MLP como las RBF son aproximadores universales (Hartman *et al.*, 1990; Park y Sandberg, 1991; Zhou, 2003; Powell, 1987), esto es, cualquier función con la suficiente suavidad puede ser escrita como la salida de una red neuronal. Al final de este capítulo se dedicará una sección a la introducción de los aproximadores universales, ilustrando alguno de los más extendidos.

1.3.3 Según el Tipo de Aprendizaje

La característica distintiva y original de las redes neuronales es el aprendizaje. A diferencia de otros sistemas tradicionales, para los que el conocimiento se expresa en forma de reglas explícitas, las redes neuronales generan sus propias reglas en el aprendizaje. Las redes neuronales aprenden de los datos, sin que sea preciso determinar una estructura para el sistema que deseamos reproducir, ni situar la distribución de probabilidad dentro de una familia concreta.

El aprendizaje de la red consiste fundamentalmente en la modificación de los pesos que conectan los nodos. Cómo aprende la red, o lo que es lo mismo, qué es lo que hace que las conexiones interneuronales se modifiquen, qué criterios se siguen, y cuándo las modificaciones son aceptadas y cuándo no, será vital a la hora de obtener buenos predictores neuronales. El proceso por el cual una red aprende se llama entrenamiento. Hay diversas clasificaciones del aprendizaje y por consiguiente de las redes, según diferentes criterios.

1.3.3.1 Redes On-Line frente a Redes Off-Line

Una primera división distingue entre redes *off line* y redes *on line*.

Las redes *off line* se caracterizan porque para realizar su aprendizaje ha de detenerse el funcionamiento de la red. Se distinguen en este tipo de redes dos etapas: una de entrenamiento y otra en que la red se dedica a predecir. Cuando la red proporciona predicciones, no se encuentra entrenando, y mientras entrena está inhabilitada para dar respuesta a nuevos datos.

Por su parte las redes *on line* tienen la característica de que entrenan con cada nuevo dato que recibe el sistema, sin necesidad de detener su funcionamiento. Los datos se modifican dinámicamente con cada nueva información.

Las primeras redes necesitarán actualizaciones periódicas, sobre todo si el proceso que se desea estudiar evoluciona con el tiempo. Pero a cambio su carácter estático durante los períodos de predicción hace más estable al sistema. Si la red se modifica constantemente con cada nuevo dato sería necesario un exhaustivo estudio para analizar la inestabilidad del sistema.

1.3.3.2 Redes con Aprendizaje Supervisado versus No Supervisado

La clasificación más usual es aquella que distingue entre redes con Aprendizaje Supervisado y con *Aprendizaje no Supervisado*. La diferencia principal radica en la existencia de un "supervisor" que controla el aprendizaje, indicando, bien hacia dónde han de modificarse los pesos, o bien si la modificación es correcta o no.

A continuación se estudian con más detalle las diferencias entre los distintos tipos de aprendizaje, para que facilitar el discernimiento entre este tipo de redes, y por tanto la clasificación.

1.3.3.2.1 Aprendizaje Supervisado

Se caracteriza por la existencia de una agente externo que conoce la respuesta que debería generar la red a partir de una determinada entrada. La salida de la red es

comparada con la respuesta deseada, y si no coinciden los pesos de las conexiones serán modificados de modo que la salida obtenida se aproxime a la deseada. La información que maneja el supervisor no es siempre la misma. Según la naturaleza de los conocimientos de los que disponga, aparecerán tres grandes clases de aprendizaje supervisado.

(i) Aprendizaje por corrección de error.

(ii) Aprendizaje por refuerzo.

(iii) Aprendizaje estocástico.

(i) Aprendizaje por corrección de error.

Es el modo más común de aprendizaje. Cada caso del conjunto de entrenamiento está constituido por las variables de entrada (que caracterizan la situación en que se encuentra el sistema) y la salida o salidas que se desean de la red (variables objetivo). El ajuste de los pesos se realizará en función de la diferencia entre los valores deseados y los que se obtuvieron en la salida de la red.

La Regla de Aprendizaje del Perceptrón, fue desarrollada por Rosenblatt (1958), y constituye el primer ejemplo de aprendizaje supervisado. Presenta el problema de que no considera de modo global el error cometido por la red al estimar el conjunto de entrenamiento. Más adelante Widrow y Hoff (1960) desarrollaron la Regla Delta, que permite conocer el error global cometido durante el entrenamiento. Estos autores aplicaron este método de entrenamiento a muchas de las redes que desarrollaron, como el ADALINE y el MADALINE. La Regla delta estaba pensada para redes constituidas únicamente por una capa de entrada y una de salida. Cuando se empezaron a diseñar redes más complejas, con una o varias capas ocultas (siempre con conexiones hacia delante) se hizo necesaria una generalización de ese algoritmo que tan buenos resultados había proporcionado. Surgió entonces la Regla Delta Generalizada. Esa regla modifica los nodos cada vez que una observación, que será elegida de modo aleatorio, es presentada a la red, y lo hace siguiendo un orden determinado, empezando por los nodos que conectan la última capa oculta con la capa de salida, y finalizando en los que unen la capa de entrada con la primera capa oculta. La Figura 1.8 detalla el esquema general del proceso de aprendizaje.

La modificación de los pesos busca disminuir la función de error hasta hallar un mínimo de esa función. Si el error es una función lo suficientemente suave, basta con buscar un mínimo local, esto es, un punto donde la derivada sea nula. Se modifican los pesos en función de la derivada del error con respecto a al peso que se desea actualizar. La amplitud de la modificación viene determinada por un paso, en general fijo durante todo el entrenamiento. Escoger adecuadamente el valor de ese paso será determinante en el éxito del entrenamiento.

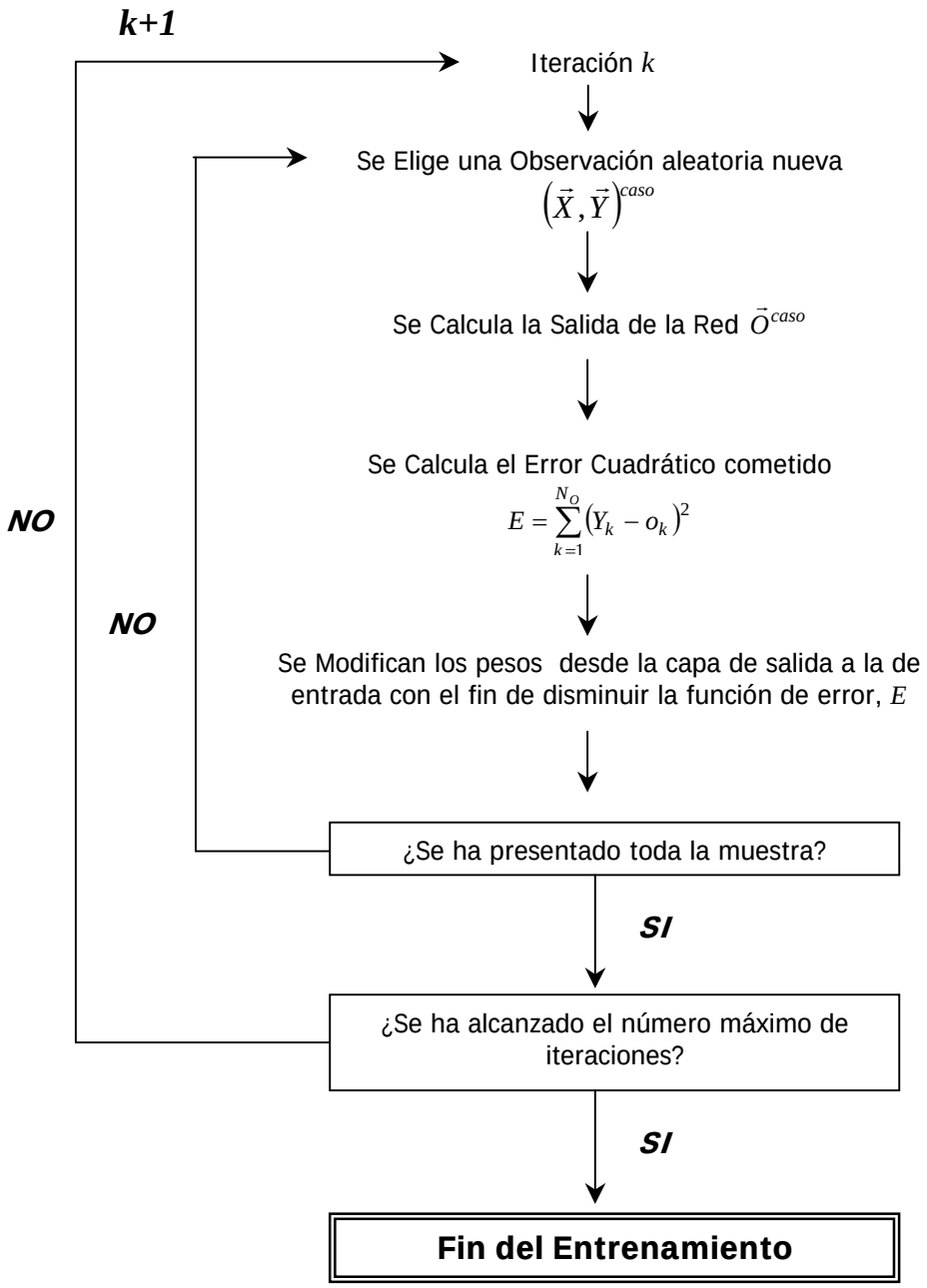

Figura 1.8. Diagrama de Flujo del Entrenamiento

Los pesos se modifican en orden inverso a su actuación en la red, pues la derivada de la función de error de los pesos de una capa dependerá de los pesos de las siguientes capas. La modificación en la iteración k-ésima será,

$$ {}^{[k]}w_{ij}^{h_{capa\,o}} = {}^{[k-1]}w_{ij}^{h_{capa\,o}} - \eta\frac{\partial E}{\partial\,{}^{[k-1]}w_{ij}^{h_{capa\,o}}} \quad , \ldots, \quad {}^{[k]}w_{ij}^{ah_1} = {}^{[k-1]}w_{ij}^{ah_1} - \eta\frac{\partial E}{\partial\,{}^{[k-1]}w_{ij}^{ah_1}} \qquad (1.22) $$

En general,

$$ {}^{[k]}w_{ij} = {}^{]k-1]}w_{ij} - \eta \frac{\partial E}{\partial {}^{[k-1]}w_{ij}} \tag{1.23} $$

La idea para su implementación se conoce como *backpropagation*. Fue desarrollada por Rumelhart, Hinton y Williams (1986) basándose en las ideas de Rosemblatt (1962).

a) Backpropagation

Consiste en tratar de minimizar una cierta función del error que comete la salida proporcionada por la red con respecto al valor deseado de la variable objetivo, por regla general proporcional al error cuadrático medio. Para ello se busca un extremo relativo de esa función, considerada como función de los pesos, esto es, un punto donde todas las derivadas parciales de la función de error respecto a los pesos se anulan.

$$ \begin{aligned} &\frac{\partial E}{\partial \omega_{ij_1}^{ah_1}} = 0; \quad con \quad 1 \le i \le N_I \\ &\frac{\partial E}{\partial \omega_{j_l j_{l+1}}^{h_l h_{l+1}}} = 0; \; con \quad 1 \le j_l \le N_{H^l}, 1 \le l \le L_H - 1 \\ &\frac{\partial E}{\partial \omega_{j_{L_H} k}^{h_{L_H} o}} = 0; \quad con \quad 1 \le k \le N_O \end{aligned} \tag{1.24} $$

El algoritmo de Backpropagation es un método clásico de entrenamiento de redes cuando se trata de en un caso de aprendizaje con supervisión. El error cuadrático medio se considera como función de los pesos, y por ello se calculan las derivadas de la función de error (fuese cual fuese) con respecto a ellos. Consiste, por tanto, en la aplicación del algoritmo de paso descendiente teniendo en cuenta el orden adecuado de cálculo de las derivadas.

Si se considera como función de error el error cuadrático medio, se aprecia que para el cálculo la derivada se necesita manejar simultáneamente todos los datos. Cuando sea necesario trabajar con todos los datos a la vez se dice que el método es *batch*. En la mayoría de los casos resultan mucho más interesantes los métodos *on-line*, puesto que los pesos se modifican con cada dato, de modo que se puede considerar la evolución del sistema. En muchas ocasiones si es necesario manejar al tiempo todos los casos del conjunto de entrenamiento, es posible obtener mejores resultados hallando el mínimo de la función de modo analítico.

El análogo 'on-line' de la regla delta generalizada responde al mismo esquema pero considerando como función de error la ecuación (1.25).

$$ E(\mathbf{W}, \vec{x}^p, \vec{y}^p) = \sum_{k=1}^{No} \left(y_k^p - o_k\left(\vec{x}^p, \mathbf{W}\right)\right)^2 , \tag{1.25} $$

siendo $(\vec{x}^p, \vec{y}^p)_{p=1}^n$ el conjunto de entrenamiento, $\mathbf{W}$ la matriz de pesos de la red.

Existen tres razones fundamentales para el uso de métodos *on-line* frente a los batch. Por una parte está la motivación biológica del aprendizaje de cada experiencia, por otra parte es importante el hecho de que su convergencia puede ser más rápida que en los métodos batch. En el caso en que el conjunto de información tenga un gran número de ejemplos muy parecidos o exactos, el promedio sobre una proporción pequeña de ejemplos proporcionará una buena aproximación a E y sus derivadas. Finalmente existe la creencia de que la introducción de ruido (la aleatoriedad del ejemplo que se presenta) hace que sea más fácil evitar mínimos locales durante la optimización.

El entrenamiento es un algoritmo iterativo, y por lo tanto requiere de un punto de partida y una regla de parada. Generalmente se parte de un conjunto aleatorio de pesos, prestando especial atención al espacio en el que se eligen, con el fin de evitar la saturación numérica de las unidades ocultas. La elección de regla de parada resulta también importante. Inicialmente se suele seleccionar como regla el detener el proceso cuando (si) el error, E, es pequeño. No siempre este es un criterio adecuado. Se han propuesto muchos métodos de parada, entre los que destacan aquellos que consisten en considerar un conjunto de validación simultáneamente al entrenamiento para el que se evalúen los resultados de la red en paralelo, de modo que se detenga el entrenamiento cuando la medida del error en el conjunto de validación empiece a crecer. Esto es peligroso, pues en muchos casos, tras un "valle", el error en el conjunto de validación crece lentamente durante un número grande de iteraciones para luego caer de pronto a una fracción pequeña de su mínimo inicial.

Los efectos del momento de parada son importantes. Si se detiene prematuramente los pesos ajustados dependerán de los de partida. Esto complica el análisis de los procesos de "parada temprana". Asimismo si te toma η demasiado grande podrá darse el caso de que en los pasos sucesivos la red se esté momento alrededor del punto óptimo (vector de pesos) donde se alcanza el mínimo, pero sin llegar a alcanzarlo. Si por el contrario se opta por un η demasiado pequeño la convergencia puede ser muy lenta. De nuevo se tienen diferentes opciones, como tomar η constante, pero pequeño (por ejemplo, $\eta = 0,008$, $\eta = 0,004$) , o bien considerar η_n una sucesión decreciente, con límite cero, pero con serie divergente. Por ejemplo, $\eta_n = 1/n$, con *n* un caso aleatorio.

Si se analiza en detalle el comportamiento de la regla de entrenamiento, se obtiene la siguiente descripción del proceso.

- *Elegir aleatoriamente los pesos iniciales.* En general los pesos se eligen de modo aleatorio en (0,1) pero que no estén demasiado cerca de estos extremos para que no se "saturen" los nodos (i.e. el valor de los pesos permanezca inamovible).

- *Elegir aleatoriamente un caso del conjunto de entrenamiento.* Se calcula el valor del nodo final con los pesos actuales., y se modifican , por mínimos cuadrados vamos minimizando el valor de los pesos.

- *Modificar los pesos*. Los pesos se modifican en la dirección en la que disminuya más rápidamente la función de error determinada.

Para ilustrar el proceso se ilustra a continuación un caso concreto, el del perceptrón multicapa, con una sola capa oculta. En este ejemplo se considera como función de error, E, el error cuadrático en los nodos de la última capa (1.25).Los pesos se modifican comenzando por la capa final y retrocediendo capa a capa hasta llegar a la primera. La regla de modificación de pesos, o regla de aprendizaje se describe a continuación. Durante la iteración m-ésima, en la última capa (capa que une los nodos de la capa oculta con los nodos de salida) los pesos se modifican según (1.26).

$$^{[m]}\omega_{jk}^{ho} = {}^{[m-1]}\omega_{jk}^{ho} - \eta \frac{\partial E}{\partial {}^{[m-1]}\omega_{jk}^{ho}}, 1 \le j \le N_H, 1 \le k \le N_O \quad (1.26)$$

Desarrollando el cálculo de la derivada se tienen las siguiente ecuaciones,

$$\frac{\partial E}{\partial {}^{[m-1]}\omega_{jk}^{ho}} = \frac{\partial E}{\partial q_k} \cdot \frac{\partial q_k}{\partial {}^{[m-1]}\omega_{jk}^{ho}} = \delta_k \cdot h_j \quad (1.27)$$

$$\frac{\partial q_k}{\partial {}^{[m-1]}\omega_{0k}^{ho}} = h_0 = 1 \quad (1.28)$$

$$\delta_k = \frac{\partial E}{\partial q_k} = \frac{\partial E}{\partial o_k} \cdot \frac{\partial o_k}{\partial q_k} = f_o'(q_k) \cdot \frac{\partial E}{\partial o_k} = -f_o'({}^{[m-1]}\omega_{0k}^{ho} + \sum_{j=0}^{N_H} {}^{[m-1]}\omega_{jk}^{ho} \cdot h_j) \cdot 2 \cdot (Y_k - o_k) \quad (1.29)$$

$$^{[m]}\omega_{jk}^{ho} = {}^{[m-1]}\omega_{jk}^{ho} + \eta \cdot h_j \cdot 2 \cdot f_o'({}^{[m-1]}\omega_{0k}^{ho} + \sum_{j_1=0}^{N_H} {}^{[m-1]}\omega_{j_1 k}^{ho} \cdot h_{j_1}) \cdot (Y_k - o_k) \quad (1.30)$$

$$^{[m]}\omega_{0k}^{oh} = {}^{[m-1]}\omega_{0k}^{oh} + \eta \cdot 2 \cdot f_o'({}^{[m-1]}\omega_{0k}^{oh} + \sum_{j_1=0}^{N_H} {}^{[m-1]}\omega_{j_1 k}^{oh} \cdot h_{j_1}) \cdot (Y_k - o_k) \quad (1.31)$$

En la primera capa, (capa que une los nodos de la capa de entrada con los nodos de la capa oculta) los pesos se modifican según (1.32); las ecuaciones posteriores detallan el proceso en más detalle.

$$^{[m]}\omega_{ij}^{ah} = {}^{[m-1]}\omega_{ij}^{ah} - \eta \frac{\partial E}{\partial {}^{[m-1]}\omega_{ij}^{ah}}, 1 \le i \le N_I, 1 \le j \le N_H \quad (1.32)$$

$$\frac{\partial E}{\partial {}^{[m-1]}\omega_{ij}^{ah}} = \frac{\partial E}{\partial g_j} \cdot \frac{\partial g_j}{\partial {}^{[m-1]}\omega_{ij}^{ah}} = \delta_j \cdot X_i \quad (1.33)$$

$$\frac{\partial g_j}{\partial {}^{[m-1]}\omega_{0j}^{ah}} = 1 \quad (1.34)$$

$$\delta_j = \frac{\partial E}{\partial g_j} = \frac{\partial E}{\partial h_j} \cdot \frac{\partial h_j}{\partial g_j} = f_h'(g_j) \cdot \frac{\partial E}{\partial h_j} =$$

$$= -f_h'({}^{[m-1]}\omega_{0j}^{ah} + \sum_{i=1}^{N_I} {}^{[m-1]}\omega_{ij}^{ah} \cdot X_i) \cdot 2 \cdot \sum_{k=1}^{N_O} (Y_k - o_k) \cdot \frac{\partial o_k}{\partial h_j} =$$

$$= -f_h'({}^{[m-1]}\omega_{0j}^{ah} + \sum_{i=1}^{N_I} {}^{[m-1]}\omega_{ij}^{ah} \cdot X_i) \cdot 2 \cdot \sum_{k=1}^{N_O} (Y_k - o_k) \cdot \frac{\partial o_k}{\partial q_k} \cdot \frac{\partial q_k}{\partial h_j} = \quad (1.35)$$

$$= -2 f_h'({}^{[m-1]}\omega_{0j}^{ah} + \sum_{i=1}^{N_I} {}^{[m-1]}\omega_{ij}^{ah} \cdot X_i) \cdot \sum_{k=1}^{N_O} (Y_k - o_k) \cdot f_o'(q_k) \cdot {}^{[m]}\omega_{jk}^{ho}$$

$${}^{[m]}\omega_{ij}^{ah} = {}^{[m-1]}\omega_{ij}^{ah} + \eta \cdot X_i \cdot 2 \cdot f_h'({}^{[m-1]}\omega_{0j}^{ah} + \sum_{i_1=1}^{N_I} {}^{[m-1]}\omega_{i_1 j}^{ah} \cdot X_{i_1}) \cdot \sum_{k=1}^{N_O} (Y_k - o_k) \cdot f_o'(q_k) \cdot {}^{[m]}\omega_{jk}^{ho} \quad (1.36)$$

$${}^{[m]}\omega_{0j}^{ah} = {}^{[m-1]}\omega_{oj}^{ah} - \eta \cdot 2 \cdot f_h'({}^{[m-1]}\omega_{0j}^{ah} + \sum_{i_1=1}^{N_I} {}^{[m-1]}\omega_{i_1 j}^{ah} \cdot X_{i_1}) \cdot \sum_{k=1}^{N_O} (Y_k - o_k) \cdot f_o'(q_k) \cdot {}^{[m]}\omega_{jk}^{ho} \quad (1.37)$$

- *Test de parada.* Tras recorrer todo el conjunto de información, modificando los pesos se comprueban los test de parada. Por regla general se establecen diversos test de parada, como en cualquier algoritmo iterativo. Algunos de los posibles controles son limitar el número máximo de recorridos del conjunto de información, que la máxima modificación de los pesos sea menor que cierta cantidad umbral predeterminada, o bien que la máxima modificación del error cuadrático medio sea menor que cierta cantidad umbral. Si no se cumple ninguno de los test seleccionados se retorna al segundo punto, recorriendo de nuevo el conjunto de información de modo aleatorio.

b) Variantes del Algoritmo Clásico

El algoritmo clásico ha sufrido muchas alteraciones. En los experimentos iniciales se le añadió el momento, y su suavidad exponencial se usó como el término de corrección.

$${}^{[m]}\omega_{ij} = {}^{[m-1]}\omega_{ij} - \eta \left((1-\alpha) \frac{\partial E}{\omega_{ij}} + \alpha \left(\Delta {}^{[m-1]}\omega_{ij} \right) \right) \quad (1.38)$$

Para acelerar la convergencia de los métodos se han propuesto muchas ideas, como, por ejemplo, elegir adaptativamente η y α para cada peso ω_{ij}.

Esta regla de backpropagation así como sus variantes presentan ciertos inconvenientes, como la facilidad con la que queda "atrapada" en mínimos locales, por tanto se hace indispensable elegir de modo apropiado los pesos iniciales a la hora de alcanzar buenos predictores. Además requiere que la función de error sea diferenciable con respecto a los pesos de la red, por lo tanto presenta inconvenientes a la hora de trabajar con funciones de activación no diferenciables, como ciertos núcleos, o la función umbral. Está claro que el algoritmo de backpropagation presenta ciertas limitaciones. Cuando comienza el entrenamiento se decanta por una dirección, y no explora otras

posibilidades, por lo que, si la elección inicial no era la adecuada, el entrenamiento seguramente acabará sin remisión en un mínimo local. Es por este y otros motivos que surgieron otros métodos de entrenamiento más flexibles, por ejemplo aquellos que hacen uso de los algoritmos genéticos. Existen muchos otros algoritmos de entrenamiento, que consisten básicamente en la búsqueda iterativa de mínimos cuadráticos no lineales. Cuando se realiza regresión con métodos de redes neuronales, y se conoce la variable objetivo o respuesta, el entrenamiento se hará empleando una red con aprendizaje supervisado.

(ii) Apredizaje por refuerzo.

En este caso el "supervisor" no conoce la respuesta adecuada que debería presentar la red, pero dispone de algún mecanismo que indica si la respuesta es buena o no. Si la respuesta era adecuada se reforzarán las conexiones que actuaron para obtener esa respuesta, y si no lo era, esas mismas conexiones se "inhibirán".

Hay un área de estudio llamada *Aprendizaje De Máquinas*, nacida en las comunidades de inteligencia artificial y ciencia computacional, en el también el objetivo radica en establecer la estructura del comportamiento a partir de los ejemplos, y las respuestas que se proporcionan para el aprendizaje son verdadero o falso.

(iii) Aprendizaje estocástico.

En esta red los cambios de los pesos no se hacen siguiendo un criterio de error, o de "buen camino", sino de modo aleatorio. El cambio de aceptará según las consecuencias que tengan los nuevos pesos en el comportamiento de la red, y en función de ciertas distribuciones de probabilidad. A cada red, de las infinitas posibles al variar los pesos, se le asignará el valor de una función potencial, como si se tratase de un cuerpo con cierta energía. Se busca el estado de máxima estabilidad, esto es, de energía mínima. Los cambios de los pesos se harán de modo aleatorio, y si el nuevo estado energético resulta ser más estable, se aceptarán los cambios. En caso de que los nuevos pesos aumenten la inestabilidad del sistema, no serán rechazados de modo inmediato, sino que se aceptarán los cambios en función de cierta distribución de probabilidades, que habrá sido determinada de antemano.

Dentro de las redes con aprendizaje supervisado también se puede establecer otro tipo de clasificación, según las variables objetivo que se consideren. De ese modo las *Redes Heteroasociativas* son aquellas que presentan variables objetivo diferentes de las variables de entrada, mientras que las *Redes Autoasociativas* tienen variables objetivo iguales a las variables de entrada.

1.3.3.2.2 Aprendizaje No Supervisado

No se dispone en este caso del supervisor que indique cómo actuar, y cuándo los cambios han de ser aceptados. Las redes con este tipo de aprendizaje no reciben ninguna señal del exterior que les indique si su salida es o no la adecuada. Las redes con aprendizaje no supervisado deberán autoorganizarse, en función de las similitudes y diferencias que presenten los datos de entrada. Diversas son las tareas que pueden realizar las redes con aprendizaje no supervisado. La más conocida es el Análisis Cluster. Este tipo de redes lo

que realizará agrupaciones de aquellos datos que presenten características comunes, esto es que estén, de algún modo, cercanos físicamente. Esta búsqueda de similitudes puede proporcionar diferentes salidas de red. Por una parte puede analizar el grado de similitud entre una nueva observación, y las presentadas anteriormente, o bien realizar un análisis cluster, estableciendo grupos o categorías, y proporcionando la categoría a la que pertenece un elemento, también puede proporcionar una salida que sea una función del espacio de características, de tal forma que las salidas de dos observaciones próximas estén cercanas entre sí.

Del mismo modo que el aprendizaje supervisado llevaba asociadas ciertas reglas de aprendizaje, (regla delta, regla delta generalizada), hay dos reglas de aprendizaje principales cuando nos referimos al aprendizaje sin supervisión, que dan lugar a dos tipos de aprendizaje.

(i) Regla de Hebb.

Se emplean principalmente cuando el objetivo consiste en estudiar la cercanía de diversas observaciones. El peso de conexión entre dos neuronas se incrementará cuando aparecen la entrada y la salida deseadas. Se considera que se ha activado una ruta, esto es, la conexión entre dos nodos, si el producto de los valores de los nodos es positivo, esto es, ambas neuronas son activas (positivas) o pasivas (negativas). Cada vez que se "activa" una ruta se incrementará el peso asociado a esa ruta.

$$\Delta\omega_{ij}^{h_{l_1} h_{l_2}} = h_i^{l_1} \cdot h_j^{l_2} \tag{1.39}$$

Si una neurona es activa y otra pasiva el peso que las une disminuirá su valor, esto es, se "inhibirá".

(ii) Aprendizaje Competitivo.

La idea de este aprendizaje se basa en que los nodos de la capa oculta han de competir entre sí, de modo que sólo uno de ellos se activa, y el resto de salidas permanecen inactivas. Una de las neuronas de la capa de salida será la vencedora, por ello esta regla recibe el nombre de "winner take all". De nuevo se trata de asociar los datos según sus características de modo que observaciones próximas den como vencedora a la misma neurona en la capa de salida.

Esta competencia se produce en todas las capas, de modo que unas neuronas actúan sobre otras "excitándolas" o "inhibiéndolas"; las neuronas que se activan entre sí están en cierto sentido asociadas, y suelen especializarse en alguna de las características de las observaciones. A la hora de aprender se tiene que tras una observación sólo se modificarán los nodos de las neuronas asociadas por activación a la salida ganadora, de modo que el peso total de la salida (la suma de los pesos asociados a ella) se redistribuya entre las conexiones activadoras.

1.4 Predicción con Redes Neuronales

1.4.1 Regresión con Redes neuronales

Las redes neuronales pueden ser entendidas como modelos generales de regresión (Haykin, 1999). Se emplean por tanto en muchas ocasiones como herramientas para predecir futuros valores de una o varias variables objetivo, que en estadística son las variables respuesta. Muchos métodos estadísticos clásicos y otros de más reciente factura han sido reescritos, no siempre de forma consciente, como redes neuronales. Esto nos da idea de lo generales que pueden llegar a ser las estructuras representadas a través de un esquema de redes neuronales, y de su clara relación con la estadística. En esta introducción se presentarán algunos modelos de regresión paramétricos y no paramétricos, que pueden ser estudiados bajo la óptica de las redes neuronales.

1.4.1.1 Regresión Lineal

1.4.1.1.1 Regresión Lineal Simple

El modelo de regresión más sencillo es la regresión lineal simple (Canavos, 2003). Se considera una pareja de variables aleatorias X, Y relacionadas linealmente. El modelo de regresión sería:

$Y = aX + b + \varepsilon$, con ε una variable aleatoria de media cero, y varianza finita. (1.40)

Luego, si se desea predecir el valor de la variable Y para $X = x$, se emplea el valor esperado de la distribución condicionada,

$$\hat{Y} = E\left[Y \middle| X = x\right] = ax + b \tag{1.41}$$

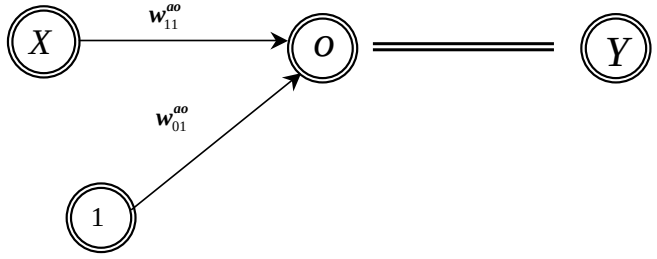

Figura 1.9. Red Neuronal para la Regresión Lineal Simple

La Figura 1.9 muestra la estructura de la red que describiría una regresión lineal simple. La salida de la red será $o = \omega_{11}^{ao} x + \omega_{01}^{ao}$, que coincide con la estructura de la predicción que proporciona un modelo de regresión lineal. Se precisa pues de un perceptrón sin capa oculta, con función de activación, la identidad, el modelo más sencillo de perceptrón, para recrear la regresión lineal simple.

1.4.1.1.2 Regresión Lineal Múltiple Multidimensional

La generalización del caso anterior (Cachero, 1996; Montgomery *et al.*, 2005)) consiste en considerar las variables explicativa y dependiente como multidimensionales, obteniéndose el modelo de regresión:

$$Y_k = \sum_{i=1}^{N_I} \omega_{ik} X_i + \omega_{0k} + \varepsilon \quad para \;\; 1 \le k \le N_O \tag{1.42}$$

La predicción sería:

$$\hat{Y}_k = E\left[Y_k \middle| X_1 = x_1, \ldots, X_{N_I} = x_{N_I} \right] = \sum_{i=1}^{N_I} \omega_{ik}^{ao} x_i + \omega_{0k}^{ao}, \;\; para \;\; 1 \le k \le N_O \tag{1.43}$$

El esquema de la red neuronal que proporciona esta misma estructura en la salida se refleja en la Figura 1.10.

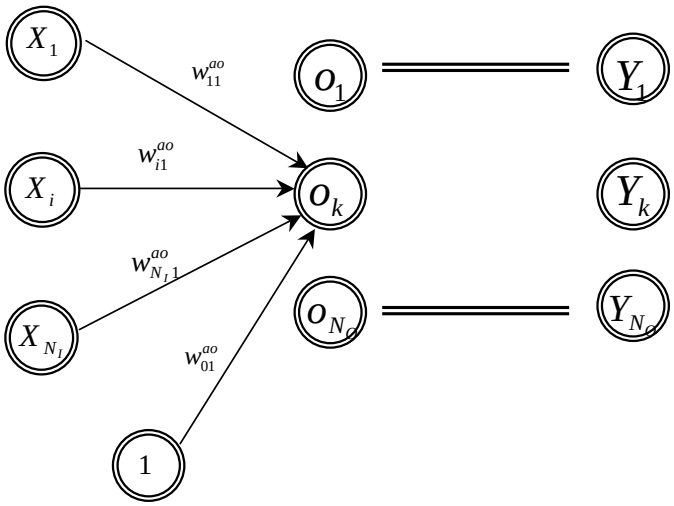

Figura 1.10. Red Neuronal para la Regresión Lineal Múltiple Multidimensional

La salida k-ésima de la red viene dada por la ecuación (1.44) y coincide con la expresión que proporciona un modelo de regresión lineal. De nuevo se emplea un modelo de perceptrón sin capa oculta, y nuevamente la función de activación es la identidad. En este caso la red habrá de tener tantos nodos de entrada como variables regresoras (N_I), y tantos nodos en la capa de salida como variables respuesta (N_O).

$$o_k = \sum_{i=1}^{N_I} \omega_{1i}^{ao} x_i + \omega_{0k}^{ao} \tag{1.44}$$

1.4.1.2 Regresión Polinómica

Una generalización natural de la regresión lineal es la regresión polinómica (Peña, 2002). Los polinomios son buenos aproximadores de una función en el entorno de un punto. Surge entonces la llamada *regresión polinómica*, que tratará de reescribir la función que relaciona la variable regresora con la variable respuesta que se desea predecir o estimar.

La predicción asociada a este el modelo de la variable dependiente k-ésima será:

$$\hat{Y}_k = E\left[Y_k \middle| X_1 = x_1, \ldots, X_{N_I} = x_{N_I} \right] = \sum_{i=1}^{N_I} \sum_{j=1}^{S} \omega_{ijk} \left(x_i \right)^j + \omega_{0k}, \;\; para \;\; 1 \le k \le N_O \tag{1.45}$$

El modo de trasladar esta idea a una red neuronal pasa por construir una capa oculta funcional. Una capa funcional, sean cuales sean las funciones que consideramos en ella, tiene como finalidad realizar transformaciones de las variables de entrada, y tienen la ventaja de que no disparan el número de parámetros, pues las conexiones que surgen entre la capa de entrada y la capa funcional tienen pesos fijos con valor 1. En ocasiones las variables $X_i, X_i^2, \ldots, X_i^S$ son muy dependientes entre sí; esto puede acarrear problemas durante el entrenamiento. Es por ello que en general es recomendable usar una base de polinomios ortogonal en la capa funcional, a fin de evitar la colinealidad que conllevan otras bases. En general cualquier función lo suficientemente suave puede ser aproximada por un polinomio, si estamos en un compacto y tomamos el grado del polinomio lo suficientemente alto, pues los polinomios constituyen lo que se ha dado en llamar un aproximador universal. La Figura 1.1,siguiendo la notación de la figura 1.3, muestra una red neuronal que refleja una regresión pilonómica de grado *S*.

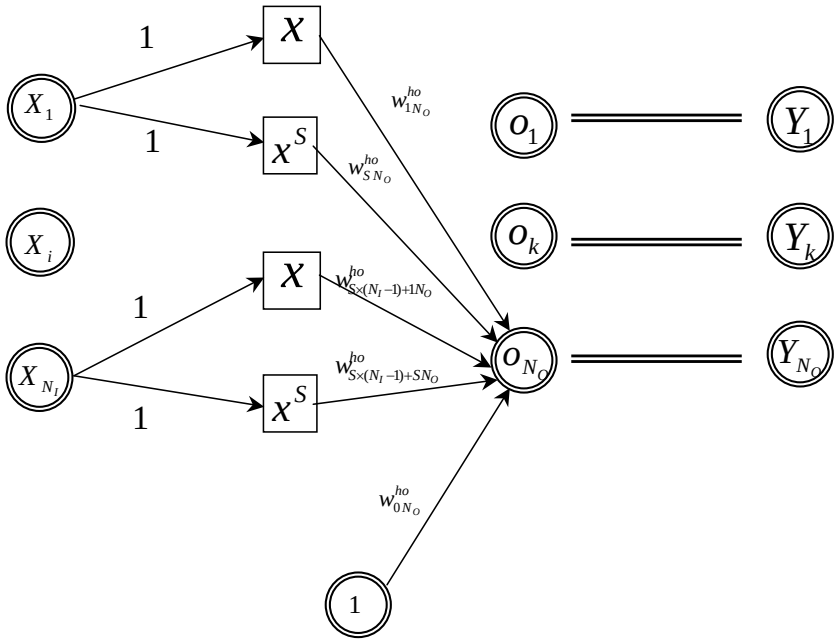

Figura 1.11. Red Neuronal para la Regresion Polinomica

La predicción que hará la red para la variable objetivo k-ésima será:

$$o_k = \sum_{j=1}^{S\cdot N_s} \omega_{jk}^{ho} X_i^j + \omega_{0k}^{ho} \text{ , con } j = (i-1)\cdot S + l, 1 \le l \le S, 1 \le i \le N_I \qquad (1.46)$$

Al analizar atentamente la red se observa que los pesos que unen la capa de entrada y la oculta son fijos, pues tomar otros daría lugar al mismo modelo al tiempo que generaría un problema de falta de especificación.

1.4.1.3 Regresión Logística

Otro modelo de regresión muy extendido es la regresión logística, que presenta múltiples aplicaciones (Artiaga *et al*., 2003). Al igual que algunos ejemplos anteriores las funciones sigmoideas también son aproximadores universales (Golberg, y Cho, 2004). En esta familia destaca la función logística.

$$\text{logist}(x) = \frac{1}{1+\exp(-x)} = \frac{\exp(x)}{1+\exp(x)},\ con\ x \in \mathbb{R} \tag{1.47}$$

Cualquier función de $\mathbb{R}^{N_I}$ en un compacto va a poder aproximarse tanto como se desee a través de una combinación de funciones logísticas de combinaciones de dichas variables. El esquema coincide con el presentado en la Figura 1.6, siendo,

$$h_j = \text{logist}\left(\sum_{i=1}^{N_O} \omega_{ij}^{ah} \cdot X_i + \omega_{0j}^{ah} \right) \quad \text{para } j = 1, \cdots, N_H \tag{1.48}$$

$$o_k = \sum_{j=1}^{N_H} \omega_{jk}^{ho} \cdot h_j + \omega_{0k}^{ho} \quad \text{para } k = 1, \cdots, N_O \tag{1.49}$$

La aproximación de la relación entre las entradas y las salidas será en principio, cuantas más funciones logísticas combinemos, esto es, cuantos más nodos constituyan la capa oculta. Pero es necesario ser cuidadosos a la hora de establecer ese número de nodos, pues un exceso de nodos derivaría en un problema de sobreestimación, casi interpolación, cuando el número de pesos se acerca al número de elementos que forman el conjunto de entrenamiento. Es posible dotar a la red de un mecanismo que elija el número de nodos de la capa oculta utilizando un conjunto de validación ajeno al de entrenamiento con el que prevenir el sobreaprendizaje.

1.4.1.4 Regresión Lineal Generalizada

La regresión polinómica constituye una generalización inmediata de la regresión lineal, en tanto en cuanto ésta es una regresión polinómica de primer grado. Otro camino por el que es posible generalizar la regresión lineal consiste en aplicar a la combinación lineal, una función determinada. De este modo se estimará la relación entre un conjunto de variables regresoras $X_1, \ldots, X_{N_I}$ y una variable respuesta Y a través de una función de la forma

$$\hat{Y} = \text{E}\left[Y \middle/ X_1 = x_1, \ldots, X_{N_I} = x_{N_I} \right] = H\left(a + b_1 x_1 + b_2 x_2 + \cdots + b_{N_I} x_{N_I} \right) \tag{1.50}$$

siendo H una función conocida.

Estos modelos reciben el nombre de Modelos Lineales Generalizados (McCullagh y Nelder, 1989; Dobson, 1990; Fox, 2008). La figura 1.12 muestra la estructura de una red que replica el esquema de la regresión lineal generalizada, en el caso de respuesta unidimensional. La extensión al caso multidimensional consistiría en diseñar tantas redes como variables se desean predecir N_O.

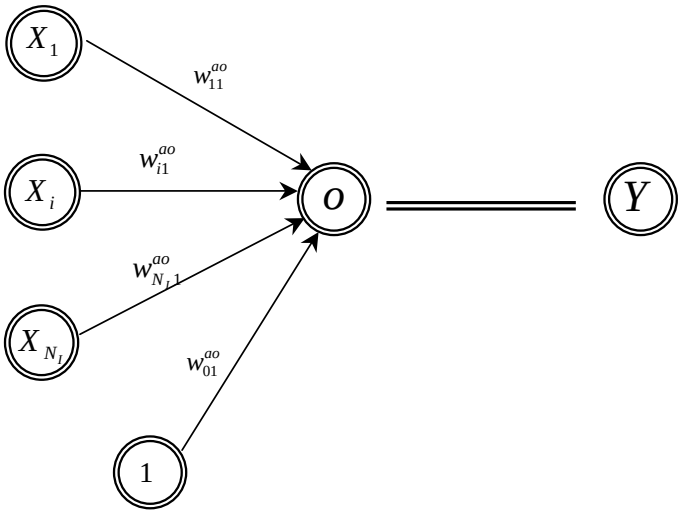

Figura 1.12. Red Neuronal para la Regresion Lineal Generalizada

Se trata de un perceptrón sin capa oculta, y con función de activación en el nodo de la capa de salida, H. De este modo, la salida que proporciona la red responde a la ecuación (1.51).

$$o = H\left(\sum_{j=1}^{N_I} \omega_{i1}^{ao} \cdot X_i + \omega_{01}^{ao}\right) \tag{1.51}$$

Aunque hasta ahora todas las redes que se habían presentado tenían como función link en los nodos de la capa de salida la identidad; este es un claro ejemplo de que no tiene que ser así, y que utilizar otras funciones link puede resultar muy útil

1.4.1.5 Regresión Aditiva Generalizada

Se considera de nuevo un modelo de regresión que incluye al anterior, con el fin de obtener resultados más generales. Se desea eliminar la restricción que conlleva la linealidad en las variables dentro de la función H. Para ello se asume que la relación entre la variable respuesta y las explicativas no responde al modelo anterior, sino que existen unas funciones $f_1, f_2, \ldots, f_{N_I}$ desconocidas, de modo que

$$\hat{Y} = \mathrm{E}\left[Y \middle/ X_1 = x_1, \ldots, X_{N_I} = x_{N_I}\right] = H\left(a + f_1(x_1) + f_2(x_2) + \cdots + f_{N_I}\left(x_{N_I}\right)\right) \tag{1.52}$$

siendo H una función conocida.

Está claro que este modelo engloba al anterior, cuando las funciones son la identidad. Estos modelos reciben el nombre de Modelos Aditivos Generalizados (Hastie y Tibshirani, 1990, Wood, 2006). Si H es la identidad corresponde al caso de los Modelos Aditivos. En este modelo han de estimarse funciones además de parámetros, lo que complica el proceso de forma notable con respecto a los modelos anteriores. Existe un método iterativo para la estimación de las funciones. Como reproducir este método en redes neuronales no sería nada sencillo, la estrategia empleada será construir N_I subredes una para cada una de las funciones, que se desean estimar. Estas redes pueden presentar arquitecturas diferentes, con diferente número de nodos en la capa oculta (tendrán un único nodo tanto en la de entrada como en la de

salida), diferentes funciones de activación, e incluso diferente número de capas ocultas. Según qué red se emplee para esas estimaciones se tendrán distintos modelos, todos ellos con el objetivo de imitar los resultados de los modelos de regresión G.A.M. La figura 1.13 muestra el diseño de la red.

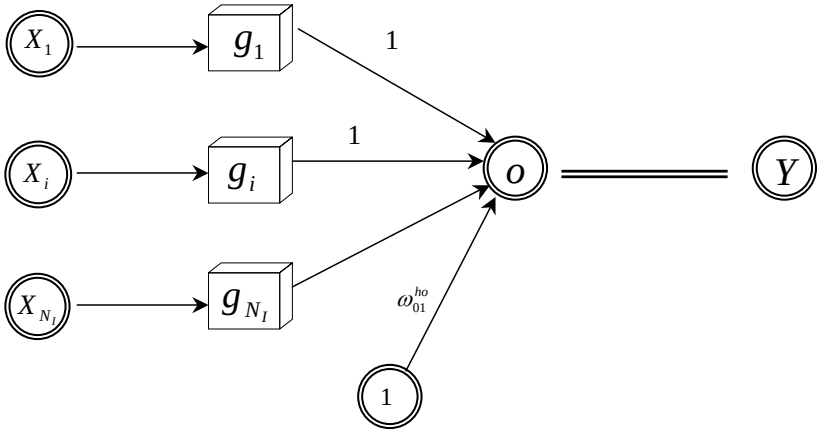

Figura 1.13. Red Neuronal para el Modelo Aditivo Generalizado

La salida de la red será:

$$o = H\left(\sum_{j=1}^{N_I} g_i + \omega_{01}^{ho}\right) \tag{1.53}$$

Los g_i se corresponden con la previsión que haría la red de $f_i(X_i)$; luego las cajas han de representar subredes. Cada una de las subredes es una red en sí misma, por lo que puede ser cualquiera de las vistas anteriormente. Como perceptrón, su diseño responderá a un esquema como el que muestra la figura 1.6, pero con un único nodo de entrada (X_i) y un único nodo de salida. Para evitar problemas de especificación se puede imponer mediante multiplicadores de Lagrange, que las variables transformadas que se obtienen de las subredes tengan media cero.

En general el tratamiento que dispensa la red a los datos será

$${}^{i}h_j = {}^{i}f_h\left({}^{i}\omega_{1j}^{ah} \cdot X_i + {}^{i}\omega_{0j}^{ho}\right) \quad \text{para } j = 1,\cdots,N_H^i, \textit{ para } i = 1,\cdots,N_I \tag{1.54}$$

$$g_i = {}^{i}f_o\left(\sum_{j=1}^{N_H^i} {}^{i}\omega_{j1}^{ho}\,{}^{i}h_j + {}^{i}\omega_{01}^{ho}\right) \text{ para } 1 \le i \le N_I \tag{1.55}$$

Este es un modelo de regresión muy general, pero tiene el inconveniente de que las variables no tienen oportunidad de "*relacionarse*" entre sí.

1.4.1.6 Regresión Projection Pursuit

Siempre buscando un modelo más general surge a la Regresión Projection Pursuit (Friedman y Stuetzle, 1981; Friedman y Tukey, 1974). La idea radica en permitir en una primera etapa la interacción de las distintas variables regresoras. El modelo supone que la relación entre las

variables independientes y la regresora depende de unas funciones desconocidas, $f_1, f_2, \ldots, f_S$ del siguiente modo:

$$\hat{Y} = a + f_1\left(b_1^1 x_1 + b_2^1 x_2 + \cdots + b_{N_I}^1 x_{N_I}\right) + \cdots + f_S\left(b_1^s x_1 + b_2^s x_2 + \cdots + b_{NI}^s x_{N_I}\right) \quad (1.56)$$

Este modelo se ve en la necesidad de estimar funciones desconocidas, tal y como sucedía en el caso anterior. La entrada de esas funciones es una combinación lineal de las variables, de modo que será necesario añadir una nueva capa oculta, previa a las subredes, donde establecer las combinaciones lineales. Se podría mantener la función H conocida que actuase globalmente, obteniendo así un modelo más general. La salida de la red será

$$o = \sum_{k=1}^{S} g_k + \omega_{o1}^{h_2 o} \quad (1.57)$$

$$\text{con} \quad m_k = \sum_{i=1}^{N_I} \omega_{ik}^{ah_1} x_i \ \ para\ 1 \le k \le S \quad (1.58)$$

y siendo g_k la estimación de $f_k\left(b_1^k x_1 + b_2^k x_2 + \cdots + b_{N_I}^k x_{N_I}\right)\ para\ 1 \le k \le S$

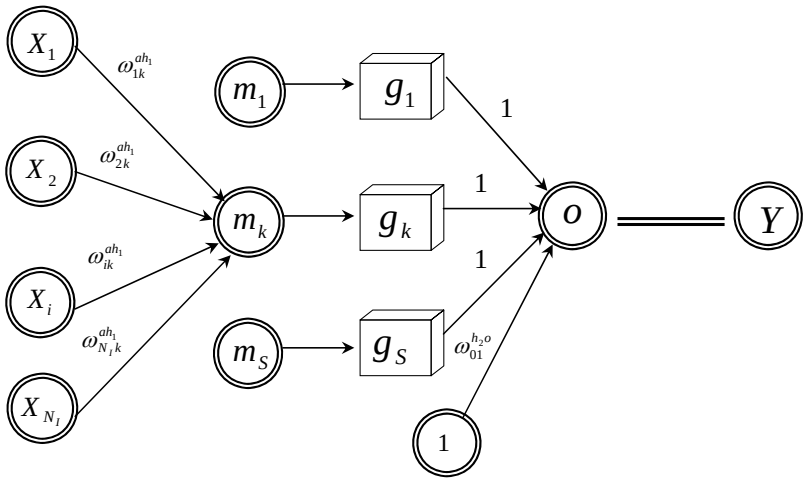

Figura 1.14. Red Neuronal para la Regresión Projection Pursuit

Se tendrán en esta ocasión S subredes como las anteriormente descritas con la única diferencia de que la variable de entrada será, para la subred k-ésima, será m_k.

1.4.1.7 Regresión G.G.A.M.

Los modelos aditivos generalizados presentaban dos restricciones principales. Una era la falta de interacción entre las variables, que se elimina en el modelo de regresión projection pursuit, y la otra era la necesidad de establecer de antemano la función H. Se puede establecer un nuevo modelo que extienda los G.A.M. eliminando esta segunda limitación. Este nuevo enfoque recibe el nombre Modelo Aditivo Generalizado General (GGAM) (Ryan, 1997; Seber y Wild, 2003). El modelo responde a la ecuación

$$\hat{Y} = \mathrm{E}\left[Y \middle/ X_1 = x_1, \ldots, X_{N_I} = x_{N_I}\right] = H\left(a + f_1(x_1) + f_2(x_2) + \cdots + f_{N_I}\left(x_{N_I}\right)\right) \quad (1.59)$$

En este modelo tanto las funciones $f_1, f_2, \ldots, f_{N_I}$ como H son desconocidas, lo que el diseño se complica notablemente. De nuevo han de ser estimadas funciones dentro de la red, pero además H presenta el inconveniente de que sus argumentos son a la vez funciones estimadas por subredes. Existe, en un enfoque estadístico clásico, un método iterativo de estimación de H, al igual que ocurría con las f_i. Según qué red, de las descritas anteriormente, se emplee para las estimaciones de las funciones f_i y de H se tendrán diferentes modelos de redes que se adaptan a los G.G.A.M.

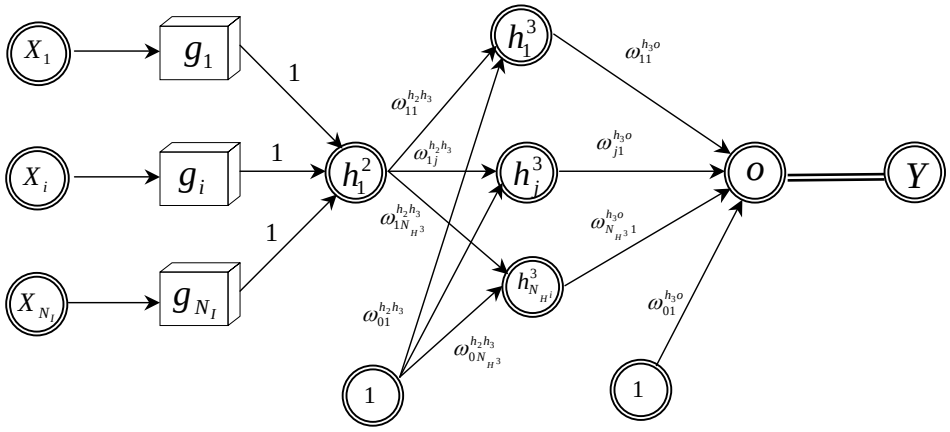

Figura 1.15. Ejemplo de Red Neuronal para los Modelos GGAM

Al analizar en detalle el comportamiento de la red se observa que los g_i se corresponden con la previsión de $f_i(X_i)$. De nuevo en la función objetivo a minimizar emplearíamos multiplicadores de Lagrange para que su media sea cero.

$h_1^2 = \sum_{k=1}^{S} g_k$ es el argumento sobre el que actúa H.

No se introduce una constante para evitar problemas de especificación a la hora de estimar esta última función. La segunda y la tercera capa ocultas, junto con la de salida constituyen la red que estima H. Este es sólo un diseño de muestra, pues existen múltiples posibilidades para la estimación de esta función, a través de una red con capa funcional, con múltiples capas ocultas, una red de base radial,...El comportamiento de las subredes ya ha sido estudiado en los subapartados anteriores. Finalmente las expresiones de la última capa oculta resultan del siguiente modo:

$$h_j^3 = f_{h^3}\left(\omega_{1j}^{h_2h_3} \cdot h_1^2 + \omega_{0j}^{h_2h_3}\right) \quad para \quad j = 1, \cdots, N_{H^3} \tag{1.60}$$

La salida de la red será, $o = f_0\left(\sum_{j=1}^{N_{H^3}} \omega_{j1}^{h_3o} \cdot h_j^3 + \omega_{01}^{h_3o}\right)$ (1.61)

1.4.1.8 Regresión Single Index Model

La restricción de establecer una función H en el modelo aparecía también en los modelos lineales generalizados (G.L.M.). Es posible, por tanto, establecer los modelos que extienden

los GLM suprimiendo el carácter fijo de H. Surgen de modo natural los Single Index Model (Stoker, 1986; Hardle and Stoker, 1989; Ichimura, 1993: Delecroix *et al.*, 2003), que responden a la siguiente relación entre las variables explicativas y la dependiente.

$$\hat{Y} = \mathrm{E}\left[Y / X_1, \ldots, X_{N_I} \right] = H\left(a + b_1 X_1 + b_2 X_2 + \cdots + b_{N_I} X_{N_I} \right) \tag{1.62}$$

siendo ahora H una función desconocida.

Será necesario establecer una subred para la estimación de H. El modelo resultante es notablemente más sencillo que el generado para los modelos G.G.A.M.

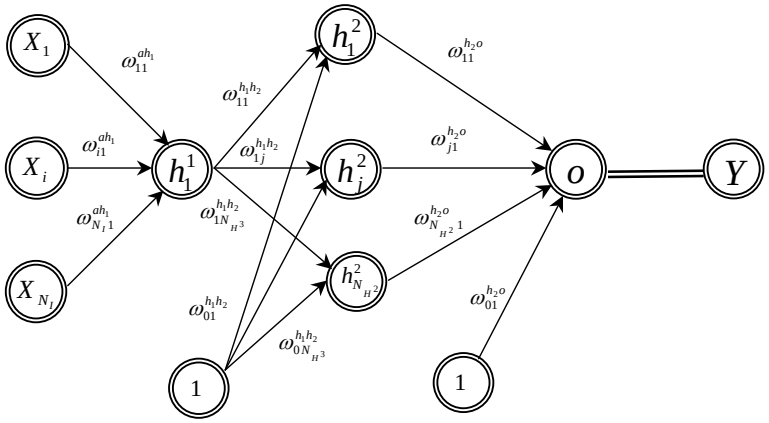

Figura 1.16. Red Neuronal para la Regresión Single Index Model

Se tiene $h_1^1 = \sum_{i=1}^{N_I} X_i$, de nuevo es el argumento de H.

$$h_j^2 = f_{h^2}\left(\omega_{1j}^{h_1h_2} \cdot h_1^1 + \omega_{0j}^{h_1h_2} \right) \quad para \quad j = 1, \cdots, N_{H^2} \tag{1.63}$$

Finalmente la salida de la red será,

$$o = f_0\left(\sum_{j=1}^{N_{H^2}} \omega_{j1}^{h_2o} \cdot h_j^2 + \omega_{01}^{h_2o} \right) \tag{1.64}$$

Analizando las expresiones (1.62) y (1.56), así como las estructuras de las redes, resulta evidente que estos modelos constituyen un caso particular de la regresión Projection Pursuit.

1.4.1.9 Regresión Tipo Núcleo

Sea $\left(\vec{X}_i, \vec{Y}_i\right)_{i=1}^{M} = \left(\left(X_i^1, X_i^2, \ldots, X_i^{N_I} \right), \left(Y_i^1, Y_i^2, \ldots, Y_i^{N_O} \right) \right)_{i=1}^{M}$ una muestra aleatoria simple. La regresión tipo núcleo (Gasser y Müller, 1984; Gasser *et al.*, 1985; Wand y Jones, 1995; Simonoff, 1996; Seber y Wild, 2003; Wasserman, 2005) responde a la idea de que, $\boldsymbol{Y}$, la variables respuesta en una nuevo punto $\boldsymbol{X}$, se puede estimar como una combinación lineal de los valores que toma en los puntos de una muestra significativa, que funcionará como los centros de las redes de base radial. La combinación lineal viene determinada por la distancia a esos puntos, merced a funciones tipo núcleo, K. En el caso de respuesta unidimensional la relación respondería a la ecuación (1.65)

$$\hat{Y}^k = \frac{\frac{1}{n}\sum_{i=1}^{M} K_h\left(\left\|\vec{X} - \vec{X}_i\right\|\right) \cdot Y_i^k}{\frac{1}{n}\sum_{i=1}^{M} K_h\left(\left\|\vec{X} - \vec{X}_i\right\|\right)} = \frac{\frac{1}{nh}\sum_{i=1}^{M} K\left(\frac{\left\|\vec{X} - \vec{X}_i\right\|}{h}\right) \cdot Y_i^k}{\frac{1}{nh}\sum_{i=1}^{M} K\left(\frac{\left\|\vec{X} - \vec{X}_i\right\|}{h}\right)} = \frac{\sum_{i=1}^{M} K\left(\frac{\left\|\vec{X} - \vec{X}_i\right\|}{h}\right) \cdot Y_i^k}{\sum_{i=1}^{M} K\left(\frac{\left\|\vec{X} - \vec{X}_i\right\|}{h}\right)} \; ; 1 \le k \le N_o \quad (1.65)$$

De este modo se observa que los pesos asociados a los distintos Y_i son 1. Se trata del predictor de *Nadaraya-Watson* (1964). Como se puede apreciar la filosofía es la misma que subyace en las redes neuronales de tipo RBF. No existe una estructura impuesta, sino que serán los propios datos los que guíen al predictor. Existen dos elementos fundamentales cuya selección resulta de vital importancia en la regresión tipo núcleo. En primer lugar está la elección de la muestra de partida. Ha de ser representativa de la relación existente entre las variables explicativas y la(s) dependiente(s). Por otra parte está el parámetro h, el llamado *parámetro ventana* (Härdle *et. al.*, 1992; Gasser *et. al.*, 1991; Härdle y Marron, 1985a, Härdle y Marron, 1985b). La correcta elección de este parámetro es imprescindible para que la regresión funcione de modo adecuado. El parámetro ventana se mantiene fijo para todos los nodos, y determina el entorno en el que influyen, o lo que es lo mismo, dado un nuevo punto, indica el tamaño del entorno por el que va a estar condicionado. Si el parámetro h es demasiado pequeño, los nodos casi no influirán en su entorno, de modo que la estimación de la variable Y podría variar notablemente entre puntos muy próximos. El caso extremo se presenta cuando la estimación interpolaría los nodos. En ese caso la estimación en nuevos puntos no sería fiable. Esta elección disminuiría el sesgo, pero aumentaría la varianza. Visualmente se obtendrían representaciones gráficas muy "nerviosas".

Si por el contrario el parámetro h es demasiado grande, el entorno donde influyen los nodos sería muy amplio, esto es, sobre un nuevo punto influirían muchos puntos, incluso aquellos muy lejanos. La estimación de Y estaría sobresuavizada, perdiéndose las particularidades locales que pueda presentar la variable respuesta. En estas circunstancias el sesgo se vería incrementado, mientras que la varianza del estimador aumentaría. El estimador sería visualmente una función muy suave en todo el dominio.

La relación de la regresión tipo núcleo con una red neuronal se establece de resulta de modo intuitivo. Esta identificación conecta el número de nodos M con el número de nodos de la capa oculta, $N_{H.}$ El resto de las equivalencias resultan inmediatas.

1.4.1.9.1 Regresión Tipo Núcleo Unidimensional Univariante

La Figura 1.17 muestra la estructura de la red que recoge la esencia de la regresión tipo núcleo en su caso unidimensional univariante (Wand y Jones, 1995; Scott, 2008). En la ecuaciones siguiente se detalla paso a paso su funcionamiento.

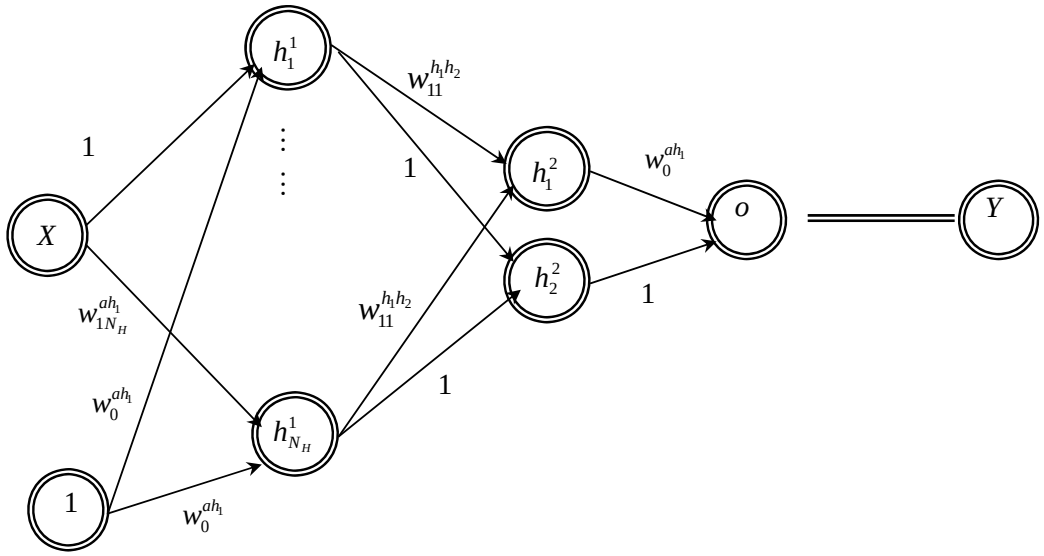

Figura 1.17. Red Neuronal para la Regresión Tipo Núcleo Unidimensional Univariante

En primer lugar resulta esclarecedor estudiar las correspondencias entre los pesos de la red y los elementos de la regresión tipo núcleo. Los sesgos o ventanas dirigidos a la primera capa oculta resultan ser todos iguales, y se corresponden con el parámetro ventana (h).

$$\omega_0^{ah_1} = h \tag{1.66}$$

El resto de los pesos que conectan la capa de entrada con la primera capa oculta corresponden a los valores de la variables regresoras en los centros, de modo que cada uno de ellos resulta asociado a un nodo de la primera capa oculta.

$$\omega_{1j}^{ah_1} = X_j, \quad para \ \ j = 1, \ldots, N_H \tag{1.67}$$

Siendo $\{(X_i, Y_i)\}_{i=1}^{M}$ el conjunto de *centros* seleccionados, que ahora se denominan $\{\vec{W}_i\}_{i=1}^{N_H} = \{\vec{W}_i^{ah}, \vec{W}_i^{ho}\}_{i=1}^{N_H}$ La función de activación de la primera capa oculta es en esta ocasión una función tipo núcleo. De este modo, las salidas de los nodos de la primera capa oculta resultan ser, en el caso unidimensional univariante,

$$h_i^1 = K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_0^{ah}}\right) = K\left(\frac{\left|X - X_i\right|}{w_0^{ah}}\right) \quad para \quad i = 1, \cdots, N_H \tag{1.68}$$

Los pesos que unen la primera capa oculta con el primer nodo de la segunda capa oculta corresponden a los valores que toma la variable dependiente en los centros.

$$\omega_{j1}^{h_1h_2} = Y_j, \quad para \ \ j = 1, \ldots, N_H \tag{1.69}$$

En el primer nodo de la segunda capa oculta responde por lo tanto a (1.70). En el segundo nodo se calcula un término relacionado con la estimación núcleo de la función de densidad, esto es, el denominador del estimador. La suma de los pesos será unitaria.

$$h_1^2 = \sum_{i=1}^{N_H} h_i^1 \cdot w_{i1}^{h_1 h_2} = \sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_0^{ah}}\right) \cdot w_{i1}^{h_1 h_2} = \sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_0^{ah}}\right) \cdot Y_i \qquad (1.70)$$

$$h_2^2 = \sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_0^{ah}}\right) = \sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_0^{ah}}\right) \qquad (1.71)$$

Finalmente la salida divide ambos términos, de modo que se obtiene (1.72), que corresponde a la estimación tipo núcleo.

$$o = \frac{h_1^2}{h_2^2} = \frac{\sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_0^{ah}}\right) \cdot w_{ii}^{h_1 h_2}}{\sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_0^{ah}}\right)} = \frac{\sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_0^{ah}}\right) \cdot Y_i}{\sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_0^{ah}}\right)} \qquad (1.72)$$

En el diagrama de la red se ha añadido la variable objetivo, con la que se desea comparar. Esta variable será fundamental a la hora de diseñar el entrenamiento, pues es la proximidad al objetivo la que guía el sentido y la magnitud de la modificación de los pesos en cada etapa.

En este caso, se desea estimar la variable respuesta, así que $O = \hat{Y}$, y por ello, la función que se minimiza en el entrenamiento es el error cuadrático medio.

$E = \frac{1}{n}\sum_{X \in T}\left(Y_X - o_X\right)^2$, con T el conjunto de entrenamiento, y *n* su cardinal (1.73)

Este es un caso de *aprendizaje supervisado,* por *corrección de error*[1]. Los únicos pesos que se modifican durante el entrenamiento son los sesgo de la primera capa oculta, que son todos iguales y coinciden con el parámetro ventana. En la metodología núcleo la elección de la ventana es un problema muy complejo que ha dado lugar a múltiples estudios teóricos. La expresión de la ventana involucra derivadas de la función que se desea estimar, que son desconocidas, y por tanto han de ser estimadas. Luego, tras un arduo estudio se obtienen valores aproximados del parámetro *h*, bien por un proceso iterativo, bien eligiendo el valor que mejores resultados entre los de una rejilla. El entrenamiento de esta red neuronal, Figura1.17, equivale a esa búsqueda de *h*.

La generalización al caso multidimensional se hace de modo natural. Sólo es necesario sustituir la distancia en el espacio unidimensional, por la correspondiente al espacio de entradas. Por su parte la generalización al caso de respuestas múltiple (N_O) se haría diseñando N_O redes, cada una con variable dependiente unidimensional.

[1] Para emplear durante el entrenamiento *la regla delta generalizada* es necesario que la función de error sea diferenciable con respecto a los pesos que se van a modificar. En este caso durante el entrenamiento sólo variará el parámetro ventana. Luego la función núcleo empleada deberá ser diferenciable. Eso no significa que no se puedan trasladar a las redes los modelos que empleen el núcleo *triangular*, o el de *epanechnikov*, entre otros, sino que habrá que buscar otro método de entrenamiento que no requiera la hipótesis de diferenciabilidad.

Si se considera la Figura 1.7, y se compara con la Figura 1.17, se aprecia que al tratar de reproducir la regresión tipo núcleo se diluye la esencia de la red, esa libertad proporcionada por los diferentes pesos. Se estudiarán algunas nuevas variantes que, conservando el espíritu de la regresión tipo núcleo, apliquen algunas de esas ventajas.

1.4.1.9.2 Regresión Tipo Núcleo. Var 1. Predictor de Nadaraya-Watson con ventana variable.

En esta primera generalización del estimador de Nadaraya-Watson (1964) el objetivo es considerar parámetros sesgo, esto es, ventanas (Fan y Gijbels, 1992; Muller y Stadtmuller, 1987), diferentes para cada uno de los centros. El resto de los parámetros permanecerán fijos durante el entrenamiento. La estructura de la red se presenta en le Figura 1.18.

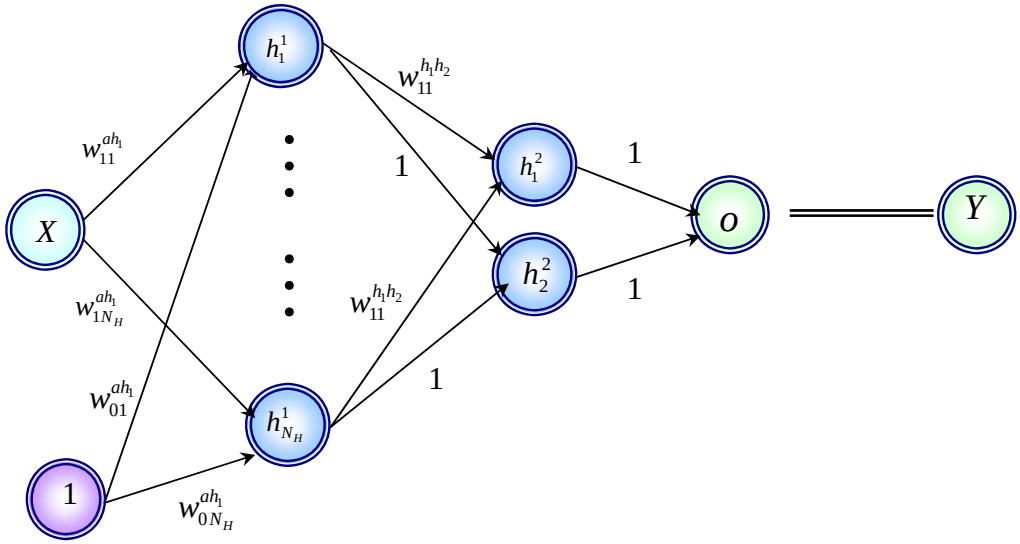

Figura 1.18. Red Neuronal para la Regresión de Nadaraya-Watson con ventana variable.

Consideramos una ventana ω_{0j}^{ah} ,que puede ser diferente para cada uno de los centros. En cualquier caso, si al final todos los centros influyen de igual modo en su entorno, el entrenamiento obtendrá ventanas similares. De nuevo durante el entrenamiento solamente se modifican los sesgos. La actuación de los nodos es la misma que en el caso anterior. Luego, la salida de la red, y por tanto la estimación de la variable respuesta responde a la expresión (1.74). Esta generalización que surge de modo intuitivo coincide con un modelo de regresión no paramética conocido, el predictor de *Nadaraya-Watson con ventana variable.*

$$\hat{Y} = o = \frac{h_2^1}{h_2^2} = \frac{\sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) \cdot w_{ii}^{h_1h_2}}{\sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right)} = \frac{\sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \cdot Y_i}{\sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right)} \qquad (1.74)$$

Esta es una primera generalización de la regresión tipo núcleo surgida de la voluntad de desarrollar las posibilidades de las redes neuronales.

1.4.1.9.3 Regresión Tipo Núcleo. Variante 2.

El siguiente paso consiste en aumentar el número de pesos que pueden ser modificados durante el entrenamiento, por ejemplo los valores de la variable respuesta en los centros.

El diagrama correspondería con la Figura 1.17, con la diferencia sustancial de que, durante el entrenamiento los pesos que alimentan el primer nodo de la segunda capa oculta son libres de ser modificados.

Otro modo de presentar esta idea se presenta en la Figura 1.19. Esta red presenta una capa oculta más, También se podría representar por una red con una capa oculta más, de modo que los centros puedan ser asignados a pesos, según (1.75).

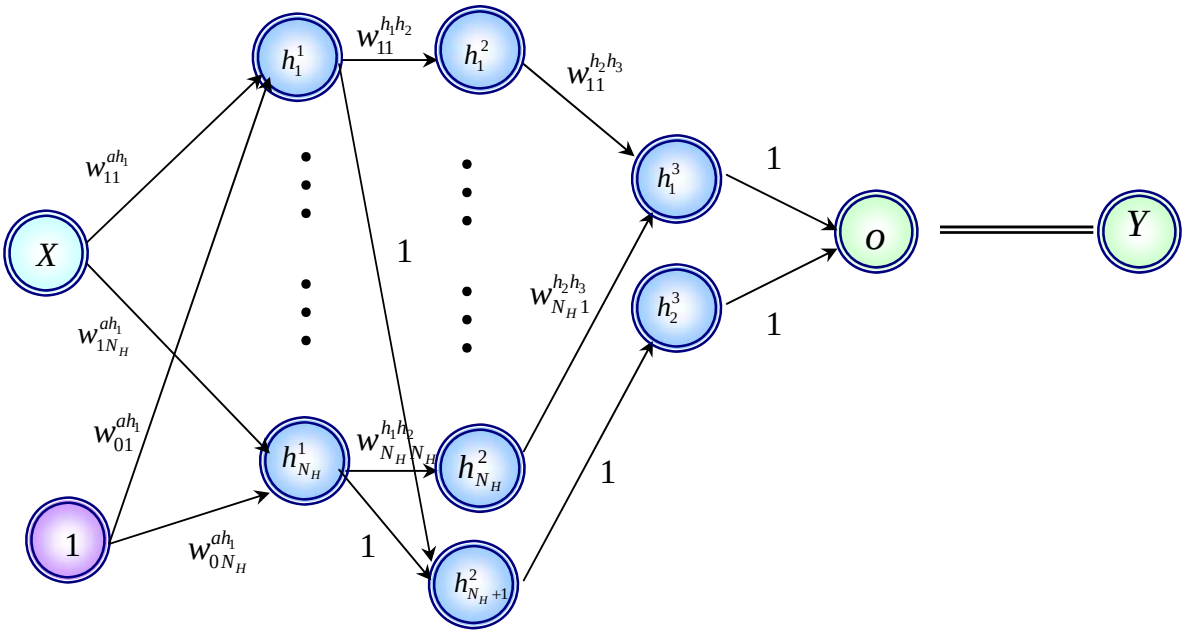

Figura 1.19. Red Neuronal para la Regresión Tipo Nucleo. Variante 2.

$$\omega_{1j}^{ah_1} = X_j,\ \omega_{jj}^{h_1h_2} = Y_j \quad para\ \ j = 1,\ldots,N_H \tag{1.75}$$

En la primera capa oculta se establece la influencia de cada centro

$$h_i^1 = K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) = K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \quad para \quad i = 1,\cdots,N_H \tag{1.76}$$

En la segunda capa oculta, por su parte se calcula el denominador, y se hacen algunos cálculos para el numerador.

$$h_i^2 = h_i^1 \cdot w_{ii}^{h_1h_2} = K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) \cdot w_{ii}^{h_1h_2} = K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \cdot Y_i,\ con\ i = 1,\ldots,N_H \tag{1.77}$$

$$h_{N_H+1}^2 = \sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) = \sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \tag{1.78}$$

En la tercera capa oculta se completa el numerador, y se incorpora la información del denominador. Está claro que la conexión podría hacerse directamente entre el último nodo de la segunda capa oculta y el nodo se la capa de salida, pero se desean diseñar redes que sólo posean conexiones entre capas consecutivas. En realidad el número de pesos no se ve incrementado, puesto que el peso entre los últimos nodos de las capas dos y tres es fijo, y tiene valor 1.

$$h_1^3 = \sum_{i=1}^{N_H} h_i^2 \cdot w_{i1}^{h_2 h_3} = \sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) \cdot w_{ii}^{h_1 h_2} \cdot w_{i1}^{h_2 h_3} = \sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \cdot Y_i \cdot w_{i1}^{h_2 h_3} \qquad (1.79)$$

$$h_2^3 = h_{N_H+1}^2 = \sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) = \sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \qquad (1.80)$$

Finalmente, la salida de la red será,

$$o = \frac{h_1^3}{h_2^3} = \frac{\sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right) \cdot w_{ii}^{h_1 h_2} \cdot w_{i1}^{h_2 h_3}}{\sum_{i=1}^{N_H} K\left(\frac{\left|X - w_{1i}^{ah}\right|}{w_{0i}^{ah}}\right)} = \frac{\sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right) \cdot Y_i \cdot w_{i1}^{h_2 h_3}}{\sum_{i=1}^{N_H} K\left(\frac{\left|X - X_i\right|}{w_{0i}^{ah}}\right)} \qquad (1.81)$$

Esta estructura permite realizar un proceso equivalente modificar los valores de la variable de respuesta, pero de modo que se puedan conservar al tiempo como pesos inalterados. De nuevo se generalizan tanto el modelo inicial como la variante anterior. Este esquema aumenta el número de pesos, pero se modifican tan sólo las ventanas y las conexiones entre la segunda capa oculta y el primer nodo de la tercera.

De este modo se solucionarán de modo natural los problemas derivados de una mala elección de centros, en el sentido de que si las repuestas en los centros son atípicas la red las modificaría en busca de otros valores más adecuados. Dicho de otro modo, se reajustan los centros tomando como base el conjunto de entrenamiento.

1.4.1.9.4 Regresión Tipo Núcleo. Variante 3.

En la práctica resulta engorroso, y más a la hora de realizar el aprendizaje, el arrastrar un denominador, pues complica, en mayor o menor medida, la expresión de las derivadas de la función de error con respecto a los pesos, y por lo tanto el proceso de entrenamiento. El único objetivo de este que tiene este término es el de normalizar la combinación lineal.

Una alternativa puede ser obviar este elemento, y considerar la expresión sin normalizar. Se pueden considerar pues, las expresiones sin normalizar tanto la versión "*fiel*" a la regresión tipo núcleo, como las *variantes 2* y *3*. La diferencia radica entonces en considerar como salida de la red el nodo en el que se calculan los numeradores, más un sesgo. De este modo se reduciría el número de capas ocultas. Considérese a modo de ejemplo la variante no normalizada asociada a la variante 1, a la que llamaremos *variante 3.2* correspondería con el diseño de la Figura 1.20

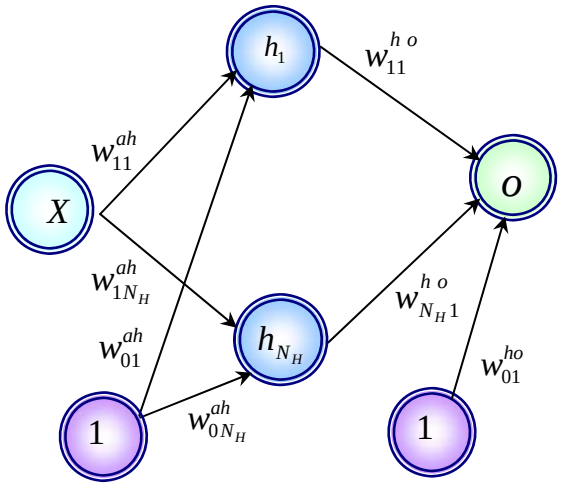

Figura 1.20. Red Neuronal para la Regresión Tipo Nucleo. Variante 3.2.

La estimación de Y responde a (1.82), de modo que nuevamente se ha llegado de modo intuitivo a un modelo conocido, el modelo de *Priestley –Chao* (1972).

$$o = \omega_{01}^{ho} + \sum_{i=1}^{N_H} K\left(\frac{x - \omega_{1i}^{ah}}{\omega_{0i}^{ah}}\right) \cdot \omega_{i1}^{ho} = \omega_{01}^{ho} + \sum_{i=1}^{N_H} K\left(\frac{x - X_i}{\omega_{0i}^{ah}}\right) \cdot Y_i \qquad (1.82)$$

Se podría pensar en mantener la expresión sin sesgo y tratar de normalizar los pesos durante el entrenamiento, empleando *multiplicadores de Lagrange* (Bertsekas, 1999) en la función de error. El problema radica en que los pesos asociados a la variable respuesta del centro *j*, (1.83), dependen del punto X en el que se esté estimando.

$$K\left(\frac{|X - X_i|}{w_{0i}^{ah}}\right) \quad para\ \ j = 1, \ldots, N_H \qquad (1.83)$$

Esto hace que no sea posible que las sumas sean unitarias para todas las observaciones del conjunto de entrenamiento.

1.4.1.10 Regresión del k-ésimo vecino más cercano

Otro método muy usual en la regresión tipo núcleo consiste en realizar una selección dentro del conjunto de centros que dependerá del el valor de X , esto es del punto en que se desee estimar la variable dependiente; el subconjunto de centros está constituido por los *k* centros con valores más próximos al punto de estimación (Hart, 1968; Dasarathy, 1991). Como antes, existe la hipótesis intuitiva de que la dependencia disminuye al aumentar la distancia. Al elegir los datos que estén más próximos se evita de algún modo la posibilidad de que en el entorno determinado por el parámetro ventana no se encuentren puntos de la muestra de centros. Pero al tiempo se generan problemas derivados de la posibilidad de que seleccionar pocos puntos en un entorno muy saturado de centros, perdiendo parte de información, o por el contrario que en un entorno despoblado sea necesario seleccionar puntos muy alejados, que pueden no contener información relevante. El papel asignado al parámetro ventana recae ahora en el parámetro *k*. EL valor de *k* está relacionado de modo directo con el valor del

parámetro ventana, de modo que la elección de *k* tiene los mismos efectos sobre el sesgo y la varianza que los de la ventana (Hastie et al, 2009).

Se considera la función *K* como constante en el entorno de radio la distancia al k-ésimo vecino más cercano. Dado un punto si se considera *r* como la distancia al k-ésimo centro más próximo, la estimación de la función de regresión será.

$$\hat{Y} = \frac{1}{k}\sum_{i=1}^{k} Y_i^X \tag{1.84}$$

con Y_i^X la variable respuesta en el i-ésimo centro más cercano a X.

Escribir este proceso en forma de red neuronal resulta sencillo. En primer es necesario transformar el conjunto de entrenamiento, para considerar en él las distancias a los *knn* puntos más próximos del conjunto de centros (que suele ser el resto del conjunto de entrenamiento), y los valores respuesta en esos puntos. Hasta este momento cuando se consideraba una observación del conjunto de entrenamiento se consideraba un par de variables aleatorias (en estos ejemplos unidimensionales). Al pasar al nuevo conjunto de entrenamiento la entrada pasará a ser (1.85), empleando la siguiente notación.

$d_i^X \left(1 \le i \le knn\right)$ es la distancia al i-ésimo punto más cercano a X

$Y_i^X \left(1 \le i \le knn\right)$ es el valor que toma la variable respuesta en dicho punto

$$\left(d_1^X, d_2^X, \ldots, d_{knn}^X, Y_1^X, Y_2^X, \ldots, Y_{knn}^X, Y\right) \tag{1.85}$$

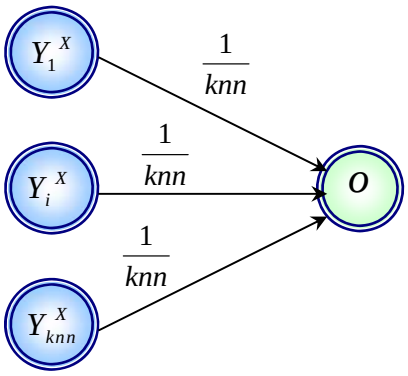

Figura 1.20. Red Neuronal la Regresión del k-ésimo vencino más cercano.

La figura 1.21 muestra el esquema de la red; se puede apreciar que no se ha empleado una red de base radial, sino un perceptrón simple, pues en realidad se está realizado una regresión lineal en los *k* puntos más próximos, pero además los coeficientes no serán modificador por la red. Es evidente la posibilidad de generar nuevas variaciones de este modelo sin más que aumentar el grado de libertad de la red en términos de entrenamiento de conexiones.

1.4.1.10.1 Variante 1.

La generalización más inmediata consiste en dar libertad a los pesos a fin de que puedan ser modificados durante el proceso de aprendizaje.

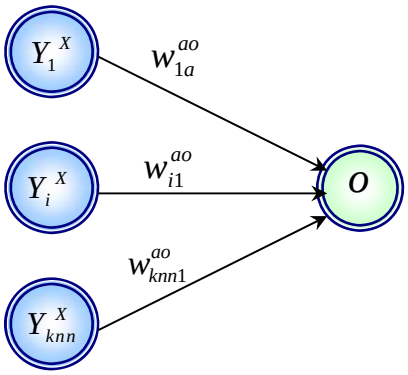

Figura 1.21. Red Neuronal la Regresión del k-ésimo vencino más cercano. Variante 1

Si la influencia de todos los puntos es la misma, los pesos tenderán durante el entrenamiento a igualarse entre sí, sumando uno. Se puede, asimismo, añadir un término de sesgo a la combinación lineal, que, en caso de ser innecesario, tendería a cero.

1.4.1.10.2 Variante 2.

Hasta este punto no se ha considerado relevante la distancia a la que se encuentran esos *k* puntos más cercanos, pero resulta intuitiva su relevancia. Se podrían considerar los vecinos como los centros, pero no es posible asociar los valores de la variable respuesta en cada centro como pesos puesto que los centros varían en función del punto de estimación, luego es necesario introducirlos como entradas. Se puede considerar la misma ventana (Variante 2.1), o ventanas distintas (Variante 2.2); además es posible considerar como modificables los valores de la variables en los puntos más próximos o mantenerlos fijos.

En este caso parece suficiente con considerar una única ventana. En otro caso se asociaría cada vecino a una ventana diferente, de modo que a un mismo nodo se le podrían asociar distintas ventanas, según el punto de estimación. Con el fin de que los pesos tengan suma unitaria se normaliza la combinación lineal.

Tanto las entradas como las salidas son consideradas entradas de la red, pudiéndose variar las respuestas a través de nuevos pesos en la red.

La Figura 1.22 muestra el esquema de la Variante 2.1. Se muestra en detalle el funcionamiento de la red a través de las ecuaciones de los diferentes nodos de las sucesivas capas.

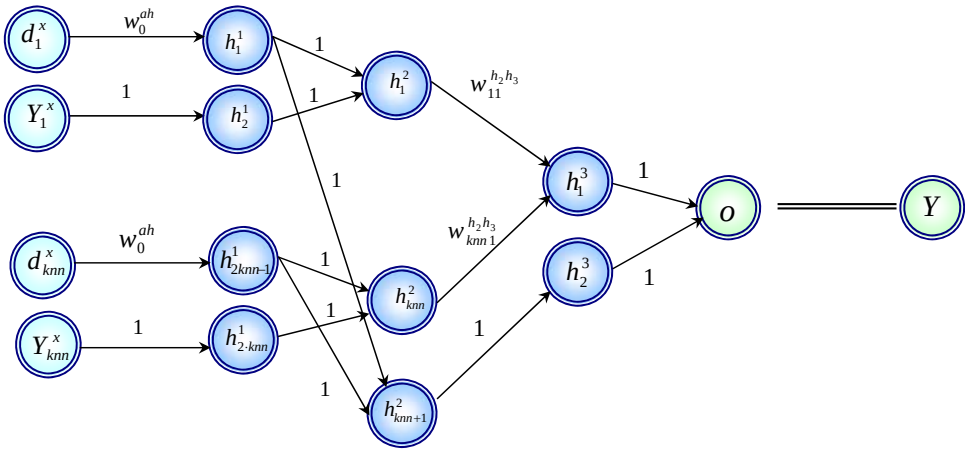

Figura 1.21. Red Neuronal la Regresión del k-ésimo vencino más cercano. Variante 2.1

El comportamiento de la primera capa oculta es,

$$h_{2i-1}^1 = K\left(\frac{d_i^x}{\omega_0^{ah_1}}\right) \quad para\ i = 1,\cdots,knn \tag{1.86}$$

$$h_{2i}^1 = Y_i^x \quad para\ i = 1,\cdots,knn \tag{1.87}$$

En la segunda capa oculta,

$$h_i^2 = h_{2i-1}^1 \cdot h_{2i}^1 = K\left(\frac{d_i^x}{\omega_{0i}^{ah_1}}\right) \cdot Y_i^x \quad para\ i = 1,\cdots,knn \tag{1.88}$$

$$h_{knn+1}^2 = \sum_{i=1}^{knn} h_{2i-1}^1 = \sum_{i=1}^{knn} K\left(\frac{d_i^x}{\omega_{0i}^{ah_1}}\right) \tag{1.89}$$

En la tercera capa oculta,

$$h_1^3 = \sum_{i=1}^{knn} h_i^2 \cdot w_{i1}^{h_2h_3} = \sum_{i=1}^{knn} K\left(\frac{d_i^X}{w_0^{ah}}\right) \cdot Y_i^X \cdot w_{i1}^{h_2h_3} \tag{1.90}$$

$$h_2^3 = h_{knn+1}^2 \tag{1.91}$$

Y finalmente, la predicción será

$$\widehat{Y} = o = \frac{h_1^3}{h_2^3} = \frac{\sum_{i=1}^{knn} K\left(\frac{d_i^X}{w_0^{ah}}\right) \cdot Y_i^X \cdot w_{i1}^{h_2h_3}}{\sum_{i=1}^{knn} K\left(\frac{d_i^X}{w_0^{ah}}\right)} \tag{1.92}$$

Si los $w_{i1}^{h_2h_3} = 1, \quad 1 \le i \le knn$, el modelo resultante es un modelo de regresión conocido, que fue desarrollado por Stone (1977). Se podrían considerar ambas opciones, normalizar o

no las combinaciones lineales, generándose así nuevas versiones, que a la vez se pueden combinar con la posibilidad de tomar una o varias ventanas.

1.5 Clasificación con Redes Neuronales

1.5.1 Consideraciones Generales

La primera tarea que fue abordada por el grupo de estudio de Rosenblatt (1958) fue la clasificación lineal de los diferentes elementos de una muestra utilizando la función umbral. Ésta es la tarea más simple de clasificación. La bondad de los resultados que proporcionó el perceptrón simple fue una de las causas del gran interés que suscitaron desde sus orígenes las redes neuronales. Los métodos de clasificación o reconocimiento de patrones supervisado responden en general al siguiente problema. Se consideran *K* clases predeterminadas, y se tiene algún modo de clasificar correctamente cualquier ejemplo presentado. Se busca un clasificador que a partir de un vector de características logre una clasificación, tan buena como sea posible. Las distintas respuestas del clasificador ante un ejemplo pueden ser:

- "El ejemplo proviene de la clase *k*"
- "El ejemplo no es de ninguna de las clases" (*Outliers*)
- "El ejemplo es demasiado complicado para mí"(*Duda*)

Tanto las respuestas de *Duda* como los Outliers tienen gran importancia en las aplicaciones prácticas. Se desea encontrar el clasificador que menos error cometa. Para establecer ese error es necesario establecer una función de pesos asociados a los errores. Ha de haber un coste de compromiso entre la tasa de duda y la tasa de error. Surgen dos cuestiones, por una parte elegir el mejor clasificador, y por otra seleccionar las características adecuadas para la clasificación.

Esta teoría se mueve en el ámbito del Reconocimiento Muestral Estadístico (Duda *et al.*, 2001; Fukunaga, 1990; Tou y González, 1974), donde no de hacen suposiciones estructurales, y toda la estructura del clasificador la proporcionan los datos (conjunto de entrenamiento).

El ser humano además de los vectores de características suele tener en ocasiones conocimientos cualitativos acerca de cierta tarea, (sólo una característica es material, o que la probabilidad de un resultado aumenta en alguna característica continua). S pueden diseñar los clasificadores para que sean *coherentes* con esta información. Se busca un clasificador, el mejor dentro de la clase que se seleccione. Resulta necesario clarificar qué es un clasificador.

Se determinan *K* clases a las que pueden pertenecer los ejemplos de un conjunto de observaciones; estas observaciones vienen determinada por un vector de características $\vec{X} \in \mathbb{X}$, con $\mathbb{X}$ espacio medible, en general $\vec{X} \in \mathbb{R}^{N_1}$.

Sean asimismo π_k las probabilidades a priori de pertenecer a la clase *k*-ésima y $p_k\left(\vec{X}\right)$ la función de densidad de $\vec{X}$ condicionado a que provenga de la clase *k*.

Un clasificador es una función asigna una observación a una clase. Un clasificador puede proporcionar *K*+2 posibles resultados o respuestas.

$$\vec{X} = \vec{x} \begin{cases} k & \text{si decide que "} \vec{x} \text{ proviene de la clase } k\text{".} \\ \boldsymbol{D} & \text{si "hay dudas", posponiéndose la decisión hasta extraer más mediciones.} \\ \boldsymbol{O} & \text{si "} \vec{x} \text{ no proviene de ninguna de las } K \text{ clases" (outliers)} \end{cases}$$

Si π_k , $p_k(\cdot)$ **son conocidas**, y siendo **C** la clase correspondiente al vector aleatorio $\vec{X}$, **C** será igual a *k* con probabilidad π_k . La tarea de clasificación consiste en estimar **C**, habiendo observado $\vec{X} = \vec{x}$.

$$\vec{X} \in \mathbb{R}^{N_I} \xrightarrow{\textit{Clasificador}} \{1, 2, \ldots K, D\} \tag{1.93}$$

Para determinar la bondad de un clasificador es necesario establecer ciertos criterios, como la probabilidad de clasificación incorrecta de los elementos de una clase *k* (1.94), y la probabilidad de duda de los elementos de una clase *k* (1.95).

$$\text{Pmc}(k) = \text{P}\{\text{C}(x) \neq \text{k}, \text{C}(x) \in \{1, \ldots, K\} | \text{C} = k\} \tag{1.94}$$

$$\text{Pd}(k) = \text{P}\{\text{C}(x) = D | \text{C} = k\} \tag{1.95}$$

Las cantidades *Pmc* y *Pd* denotan la probabilidad de clasificación incorrecta incondicional y la probabilidad de duda.

$$\text{Pmc} = \text{P}\{\text{C}(x) \neq \text{C}, \ \text{C}(x) \in \{1, \ldots, K\}\} \tag{1.96}$$

$$\text{Pd} = \text{P}\{\text{C}(x) = D\} \tag{1.97}$$

La forma usual de formalizar buen criterio es a través de una función de pérdida.

$L(k,l) \equiv$ pérdida cometida eligiendo *l* si la clase real es **C** = *k*.

Esta función ha de cumplir ciertas condiciones; en primer lugar

$L(k,k) = 0, \forall\ k \in \{1, 2, \ldots, K\}$ y se puede solicitar también que

$L(k,l) \in \mathbb{R}^+, \forall\ k, l \in \{1, 2, \ldots, K\}$

$L(k,D) = d, \forall\ k \in \{1, 2, \ldots, K\}$, con $d \geq 0$

En el caso en que todas clasificaciones incorrectas tengan la misma gravedad, entonces una elección razonable sería

$$L(k,l) = \begin{cases} 0 \ \textit{si}\ l = k \ \ (\textit{decisión correcta}) \\ 1 \ \textit{si}\ l \neq k, \textit{con}\ l \in \{1, 2, \ldots, K\} \ \ (\textit{decisión incorrecta}) \\ d \ \textit{si}\ l = D \ \ (\textit{dudamos}).\ \textit{con}\ d \geq 0 \end{cases} \tag{1.98}$$

En ocasiones el error Cuando el error puede causar dificultades o incluso peligro, se emplea con un valor de *d* muy elevado. Esta constante actúa como umbral de seguridad, y puede ser especificada por el que emplee el clasificador. Si se elige una clase al azar (no D) el valor esperado de la función de pérdida sería $1\text{-}1/K$. Por ese motivo si *d* es mayor que esa cantidad elegir la duda sería tan caro que la elección D jamás sería usado.

Pero en general no todos los errores tienen las mismas consecuencias, por lo que tampoco suelen tener la misma penalización. Se considera como función de riesgo al clasificar con C a la pérdida esperada, considerando la función de pérdida como función de la clase *k* desconocida. De este modo:

$$\begin{aligned} R(\mathrm{C},k) &= E\left[L\left(k,\mathrm{C}\left(\vec{x}\right)\right)\middle|\mathrm{C}=k\right] = \\ &= \sum_{l=1}^{K} L(k,l)\,P\left(\mathrm{C}\left(\vec{x}\right)=l\middle|\mathrm{C}=k\right) + L(k,D)\,P\left(\mathrm{C}\left(\vec{x}\right)=D\middle|\mathrm{C}=k\right) = \\ &= \mathrm{Pmc}(k) + d\cdot\mathrm{Pd}(k) \end{aligned} \tag{1.99}$$

El riesgo total es la pérdida total esperada, considerando tanto la clase, *k*, como $\vec{X}$, aleatorios.

$$R(\mathrm{C}) = E\left[R(\mathrm{C},k)\right] = \sum_{k=1}^{K}\pi_k\cdot\mathrm{Pmc}(k) + d\cdot\sum_{k=1}^{K}\pi_k\cdot\mathrm{Pd}(k) \tag{1.100}$$

En general se buscan clasificadores que minimicen el riesgo total, o a buscar la clase que maximice su probabilidad condicionada. Aplicando la función de pérdida (1.98) a $\vec{x}$, se tiene,

$$P\left(k\middle|\vec{x}\right) = P\left(\mathrm{C}=k\middle|\vec{X}=\vec{x}\right) = \frac{\pi_k p_k\left(\vec{x}\right)}{\sum_{l=1}^{K}\pi_l p_l\left(\vec{x}\right)},\ \forall k\in\{1,2,\ldots,K\} \tag{1.101}$$

Para una función de pérdida L más general el clasificador sería:

$$\mathrm{C}\left(\vec{x}\right) = \begin{cases} k \ \textit{si}\ k = \underset{l\in\{1,2,\ldots,K\}}{\arg\min}\sum_{j=1}^{K} L(j,l)\,P\left(j,\vec{x}\right) < d \\ D\ \textit{en otro caso} \end{cases}$$

Este clasificador óptimo se llama Regla de Bayes (Berger, 1985)

Si $p_k(\cdot)$ **es desconocida,** se hace necesario estimarla; en ese punto entra en juego la inferencia estadística y por tanto surge la posibilidad de aplicar redes neuronales. El objetivo consiste en estimar esas probabilidades como función de las observaciones. Todos los métodos analizados en la sección previa tratan de estimar una función que dependiente de unas entradas, como en este caso, por lo que se puede trasladar cualquier modelo anterior al ámbito de la clasificación.

La diferencia radica en que la función de error es el riesgo total, pues no se dispone de una variable objetivo (en ningún caso se conoce el valor de las densidades que deseamos estimar). A la hora de modelizar las densidades de las clases y las probabilidades condicionadas se presentan dos posibles enfoques: el Modelo Ejemplarizante y el Modelo de Diagnóstico. Ambos

dan un modelo de la densidad conjunta $p(\vec{X},\mathrm{C})$ de un ejemplo aleatorio $(\vec{X},\mathrm{C})$ vector de características y clasificación.

En el Modelo Ejemplarizante el interés se centra en estimar $p_k(\vec{X})$. Se tiene que $p(\vec{X},\mathrm{C}) = \pi_{\mathrm{C}} \cdot p_{\mathrm{C}}(\vec{X})$, siendo π_{C} las probabilidades a priori. Cuando son desconocidas se suelen estimar a partir de la proporción de casos de cada clase que presenta una muestra aleatoria simple.

En el Modelo de Diagnóstico el interés se centra en las probabilidades a posteriori $p(\mathrm{C}|\vec{X})$, puesto que $p(\vec{X},\mathrm{C}) = p(\mathrm{C}|\vec{X}) \cdot p(\vec{X})$.

Una forma sencilla de clasificar consiste en elegir la clase que maximiza $p(k|\vec{X})$, siempre que esta probabilidad supere un umbral. En caso contrario la respuesta del clasificador será la duda. El procedimiento de las redes neuronales para clasificar consistirá en la estimación de la densidad de las diferentes clases con diferentes subredes para, a continuación, escoger la clase con mayor probabilidad condicionada, si supera un umbral. El hipotético vector objetivo será una N_K-upla binaria, con un único 1, ocupando la posición asignada a la clase a la que pertenece.

1.5.2 Métodos Clásicos

1.5.2.1 Análisis Discriminante Lineal

El problema más sencillo de clasificación consiste en la clasificación en dos clases. Un enfoque sencillo consiste en intentar separar geométricamente esas dos clases a través de un hiperplano. Denotando por *1* la pertenencia al grupo *A*, y *0* la pertenencia al grupo *B*, resulta que el Análisis Discriminante Lineal (Lachenbruch, 1975; Peña, 2002) coincide con un perceptrón simple, con función de activación umbral (Figura 1.4). Este problema se puede extender a varias clases linealmente separables.

1.5.2.2 Análisis Discriminante Flexible

De nuevo el objetivo es discernir si una observación pertenece o no a una clase, pero se desea evitar la limitación impuesta por la linealidad del perceptrón simple. Añadiendo una nueva capa oculta (o varias) se logra dar más flexibilidad a la forma de la frontera que separa ambas clases. Según se seleccione una red logística, polinómica, núcleo, o se imite cualquiera de los modelos expuestos en regresión se tendrá un análisis discriminante diferente.

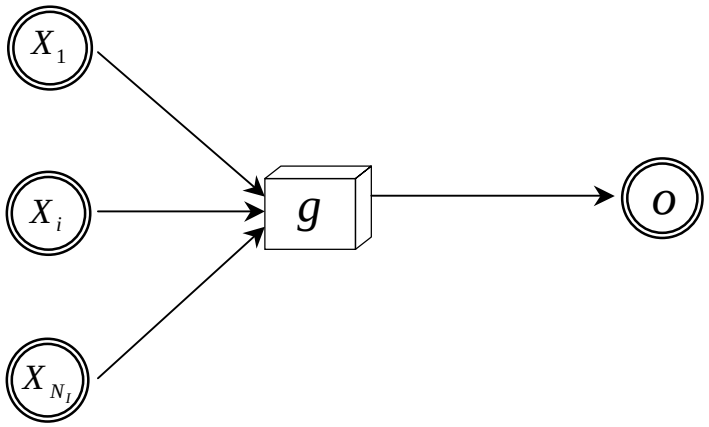

Figura 1.22. Red Neuronal para el Análisis Discriminante Flexible

Siendo g una subred que reproduce cualquiera de los métodos expuestos con anterioridad.

1.5.3 Métodos de Clasificación No Paramétricos

1.5.3.1 Estimación de la Densidad Tipo Núcleo

La estimación de la densidad tipo núcleo (Silverman, 1986; Scott, 1992; Cao *et al.*, 1994) responde a la idea de que, siendo $Y \in \{0,1\}$ el indicador de la pertenencia a un grupo, la probabilidad de que un nuevo punto pertenezca al conjunto se puede estimar como una combinación lineal de los valores que toma una función núcleo, aplicada a las distancias de ese punto a unos centros significativos de esa clase. De nuevo se emplean funciones tipo núcleo, de las que ya se señaló que suelen ser simétricas, estar acotadas e integrar uno. Esta última propiedad hará que los estimadores basados en ellas que se van a emplear también integren uno, algo fundamental en un estimador de una función de densidad. La estimación en el caso multidimensional responde a la expresión (1.102), mientras que para el caso unidimensional basta con sustituir la norma por el valor absoluto.

$$\hat{Y} = \frac{1}{n}\sum_{i=1}^{M} K_h\left(\left\|\vec{X} - \vec{X}_i\right\|\right) = \frac{1}{nh}\sum_{i=1}^{M} K\left(\frac{\left\|\vec{X} - \vec{X}_i\right\|}{h}\right) \qquad (1.102)$$

Esta expresión resulta similar a la que aparecía en el cociente de la estimación de la regresión tipo núcleo. De nuevo no se supone una estructura para la función de densidad, ni la pertenencia de la variable a ninguna familia de funciones de distribución, salvo a la de funciones absolutamente continuas. Se presentan las mismas elecciones que era necesario tomar en la estimación de la regresión. Es de vital importancia que los puntos elegidos como significativos de la densidad realmente lo sean, en particular de las diferentes densidades pues se aborda ahora un problema de clasificación. De nuevo se considerará solamente un parámetro, el parámetro ventana, h, de modo que se conserva todo lo comentado anteriormente referente a este parámetro, tanto su especial interés, como las consecuencias que ya señaladas, derivadas de una mala elección (Jones, *et al.* 1996; Sheather y Jones, 1991; Hall *et al.*, 1991).

La figura 1.23 muestra la traducción a redes neuronales del estimador (1.102). Los sesgos dirigidos a la primera capa oculta que son todos iguales y se corresponden con el *parámetro ventana* (h). El resto de los pesos que conectan la capa de entrada con la primera capa oculta corresponden a los valores de las variables en los centros, cada uno de ellos asociado a un nodo.

$$\omega_0^{ah} = h \tag{1.103}$$

$$\omega_{1j}^{ah} = X_j, \quad para\ \ j = 1,\ldots,N_H \text{ , esto es , } \left\{\vec{X}_i\right\}_{i=1}^{N_H} = \left\{\vec{W}_i^{ah}\right\}_{i=1}^{N_H} \tag{1.104}$$

El comportamiento de la red será,

$$h_j = K_{\omega_0^{ah}}\left(\left|X - \omega_{1j}^{ah}\right|\right) = \frac{1}{\omega_0^{ah}} K\left(\frac{\left|X - \omega_{1j}^{ah}\right|}{\omega_0^{ah}}\right),\ para\ \ j = 1,\ldots,N_H \tag{1.105}$$

$$\hat{Y} = o = \frac{1}{n}\sum_{j=1}^{N_H} K_{\omega_0^{ah}}\left(\left|X - \omega_{1j}^{ah}\right|\right) = \frac{1}{N_H \omega_0^{ah}}\sum_{j=1}^{N_H} K\left(\frac{\left|X - \omega_{1j}^{ah}\right|}{\omega_0^{ah}}\right) \tag{1.106}$$

Para generalizar esto al caso multidimensional basta con cambiar a una distancia en un espacio multidimensional.

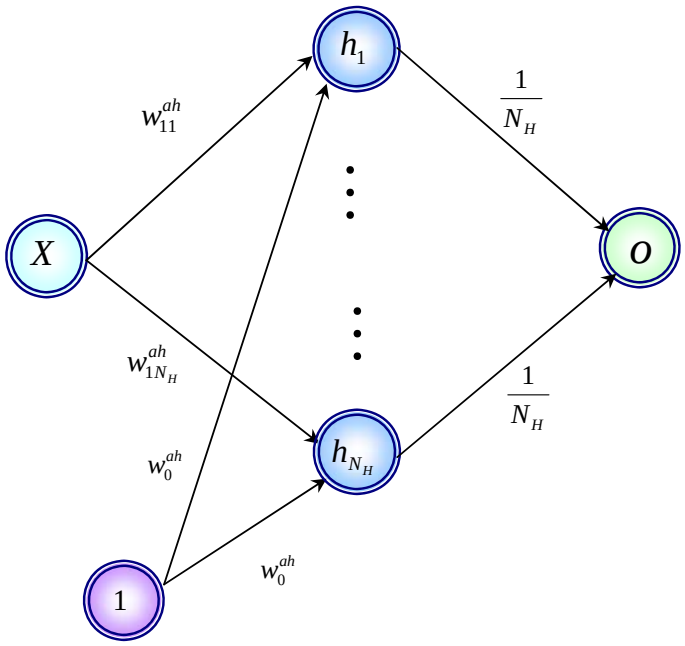

Figura 1.23. Red Neuronal para la Estimación de la Densidad Tipo Núcleo Unidiomensional

1.5.3.1.1 Estimación De La Densidad Tipo Núcleo. Variante 1.

Si se considera la posibilidad de que las ventanas puedan ser diferentes para cada uno de los centros, se generaliza el caso anterior; es posible que si todos los nodos influyen de igual modo en su entorno lleguen a tener todos la misma ventana asociada. De nuevo podemos considerar la opción de variar las ventanas según el centro que se considere.

La conexión ω_{0j}^{ah} puede ser diferente para cada uno de los centros.

De este modo la estructura del estimador obtenido se muestra en la figura 1.24.

$$\hat{Y} = o = \frac{1}{n}\sum_{j=1}^{N_H} K_{\omega_{0i}^{ah}}\left(\left|X - \omega_{1j}^{ah}\right|\right) = \frac{1}{N_H}\sum_{j=1}^{N_H}\frac{1}{\omega_{0j}^{ah}} K\left(\frac{\left|X - \omega_{1j}^{ah}\right|}{\omega_{0j}^{ah}}\right) \tag{1.107}$$

Analizando su forma resulta similar *estimador núcleo de banda variable* (Salgado-Ugarte y Pérez-Hernández, 2003; Wu *et al.*, 2007) pero en ese estimador las ventanas h_i coinciden con la distancia al k-ésimo punto más próximo. Por ello esta variante generaliza a aquel estimador.

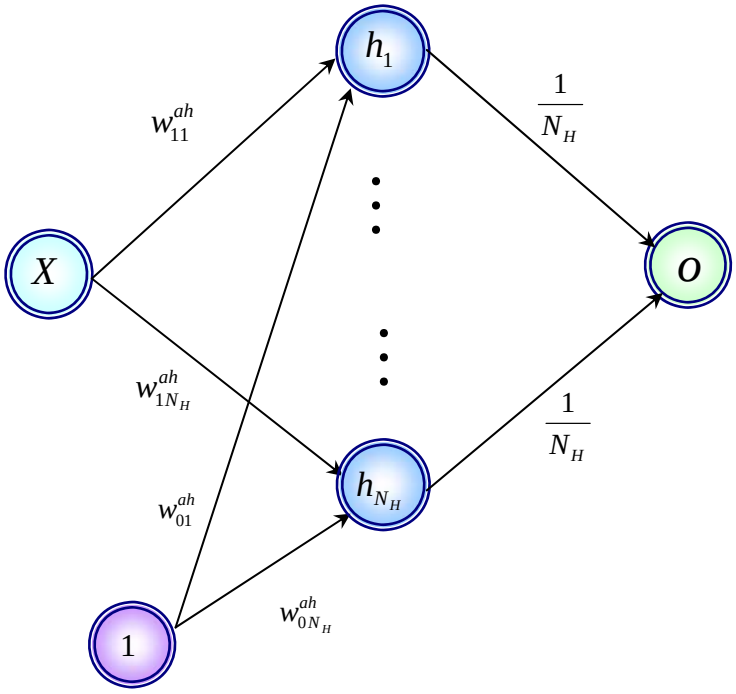

Figura 1.24. Red Neuronal para la Estimación de la Densidad Tipo Núcleo Unidiomensional. Variante 1.

Existen claramente más posibilidades para seguir generalizando el estimador, puesto que hay otros pesos que son modificables, los que llegan a la capa de salida.

1.5.3.1.2 Estimación De La Densidad Tipo Núcleo. Variante 2.

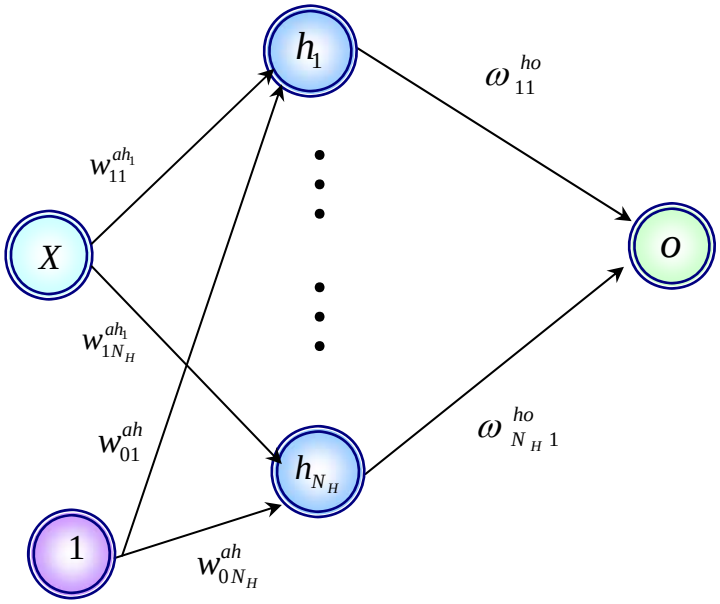

Figura 1.25. Red Neuronal para la Estimación de la Densidad Tipo Núcleo Unidiomensional. Variante 2.

La idea consiste en permitir a la red generar combinaciones lineales diferentes, y no sólo el promedio de los datos; esto genera un aumento en el número de pesos.

El diagrama responde a la figura 1.25. Si se desea mantener el hecho de que los pesos sumen uno, basta con añadir términos cuando sea preciso a la función de error empleando multiplicadores de Lagrange.

1.5.3.1.3 Estimación De La Densidad Tipo Núcleo. Variante 3.

Parece en principio que ya no es posible aumenta los grados de libertad de la red. Una posible alternativa consiste en modificar las observaciones de cada clase, i.e., los X_i durante el proceso, (al estilo del análisis cluster (Kaufman y Rousseeuw, 1990). De este modo no se arrastrarían los errores derivados del hecho de que la muestra para una o varias clases pudiera ser poco acertada.

La nueva idea consiste por tanto, en hacer variables los centros. Se señaló en su momento la necesidad que tiene la *regla delta* generalizada de que el estimador resultante sea diferenciable con respecto a los pesos. Resulta claro que las distancias no suelen ser cumplir esta propiedad, por lo que sería necesario emplear otra regla de entrenamiento menos restrictiva. El esquema se correspondería con la Figura 1.25, pero ahora todos los parámetros son variables, lo que hace aún más general el caso. Se podría rescribir la red de modo similar a como se hizo en la sección 1.4.1.9.3, en particular como muestra la Figura 1.19, para modificar las variables respuestas, esto es, crear una nueva capa oculta, que multiplique los X_i por pesos, con el fin de proporcionar libertad al conjunto de centros a la tiempo que se conservan los originales.

1.5.3.2 Estimación de la Densidad del k-ésimo Vecino más Cercano

La idea de este estimador (Cover y Hart, 1967; Wand y Jones, 1995) radica en observar la amplitud del entorno necesaria para registrar *k* centros en un entorno de un punto. El *k*, al igual que le tamaño de la ventana dependerá del tamaño de la muestra de centros. En el intervalo $\left[x-r, x+r\right]$ la probabilidad de que se encuentre algún centro responde a una binomial de parámetros n (N_H) y p (1.108). El valor esperado responde pues a (1.109).

$$p = P\left(x - r < X < x + r\right) \tag{1.108}$$

$$nP\left(x - r < X < x + r\right) = n\int_{x-r}^{x+r} f(t)dt \tag{1.109}$$

Esta esperanza puede se aproximada por $2nrf(x)$.Un estimador natural responde a (1.110).

$$\hat{f}(x) = \frac{número\ de\ centros\ en\ \left(x - r, x + r\right)}{2nr} \tag{1.110}$$

En general se varía r y se cuenta el número de casos presenta en el intervalo, pero en el caso de la estimación del *k*-ésimo vencido más cercano se fija el número de observaciones *k* (knn) y en función de ellos varía r. El estimador será (1.111), siendo r la distancia a su vecino *k* más próximo.

$$\hat{f}(x) = \frac{k}{2nr} \tag{1.111}$$

A la hora de trasladar el estimador al espacio de las redes neuronales, de nuevo resulta necesario transformar el conjunto de entrenamiento inicial, considerando las distancias a los centros, que suelen ser el resto de los puntos del conjunto de entrenamiento. El nuevo conjunto tiene como variables explicativas las distancias a los *knn* puntos más próximos, y como variables respuestas las asociadas a esos puntos. La figura 1.26 muestra la simplicidad de la red que calcula el estimador de la densidad.

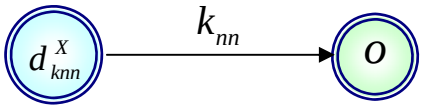

Figura 1.26. Red Neuronal para la Estimación de la Densidad Tipo Núcleo Unidimensional. Variante 2.

$$o = \frac{knn}{2nd_{knn}^{X}} \tag{1.112}$$

La estimación que proporciona la red será (1.112). En realidad no se están empleando herramientas propias de redes neuronales. Se podrían intentar emplear las características de la red para tratar de seleccionar el *k.* Un posible modo de hacerlo sería repitiendo el mismo esquema en paralelo, desde un valor de *k* mínimo hasta otro máximo, de modo que el que mejor clasifique en esa clase sea el elegido. Es en cierto modo, un mecanismo de tanteo.

1.6 Otros Métodos de Análisis de Datos

1.6.1 Análisis Factorial

Tanto el Análisis Factorial como el Análisis de Componentes Principales (Peña, 2002; Cuadras, 1981), tienen como objetivo reducir la dimensión del problema. El Análisis Factorial (Harman, 1976; Scofier y Pagès, 1992) parte de la suposición implícita de que los datos han sido generados de una forma particular, a partir de ciertos factores mediante combinaciones lineales, por lo que los resultados obtenidos podrán además ser objeto de interpretación. Da lugar, por lo tanto, a un modelo de inferencia. Los factores serán a su vez combinaciones lineales de las variables originales, y son aquellas combinaciones que mejor las reproducen. El procedimiento consiste en el cálculo de combinaciones lineales, tantas como factores se deseen, de modo que al recalcular por combinaciones lineales las variables originales, se produzca el menor error posible. De este modo las combinaciones intermedias serán los factores deseados. Así sabremos que las combinaciones intermedias son los factores que buscamos. El proceso por lo tanto difiere del seguido en los ejemplos anteriores. No se calcula directamente el error entre el factor y nuestra estimación, sino que se requiere un paso extra que indique el camino adecuado. Este es un claro ejemplo del aprendizaje por refuerzo.

La capa oculta estará constituida por tantos nodos como factores se deseen estimar, N_H, recordando que serán a lo sumo tantos como variables de entrada. La red asimismo carecerá de sesgos.

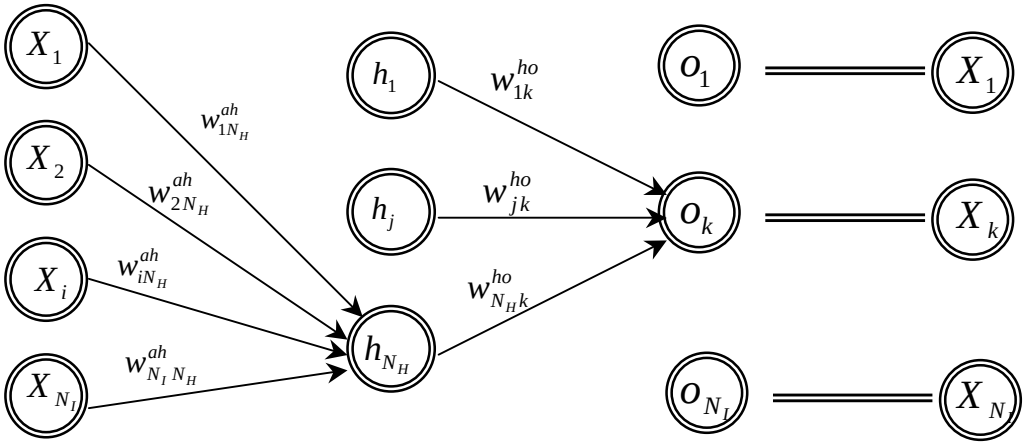

Figura 1.27. Red Neuronal para el Análisis Factorial

Las expresiones de las estimaciones de los factores se muestran en la ecuación (1.113), mientras que las salidas de la red se muestran en (1.114).

$$h_j = \sum_{i=1}^{N_I} \omega_{ij}^{ah} X_i, \; para \, 1 \le j \le N_H \tag{1.113}$$

$$o_k = \sum_{j=1}^{N_H} \omega_{jk}^{ho} \cdot h_j \quad para \; k = 1, \cdots, N_I \tag{1.114}$$

Las salidas de la red se compararán con las entradas X_k correspondientes, a través de la función de error cuadrático, de modo que los factores se estiman en la capa oculta.

1.6.2 Análisis de Componentes Principales

El Análisis de Componentes Principales (Hair *et al.*, 1999; Aluja Banet y Morineau, 1999) tiene como objetivo reducir la dimensión, conservando la mayor cantidad de información posible, esto es, explicando la mayor variabilidad posible. La Primera Componente Principal será entonces la combinación lineal de las variables, que tenga mayor varianza. La Segunda Componente Principal es una variable aleatoria, de nuevo combinación lineal de las originales, independiente de la anterior, en particular ortogonal a ella, de varianza máxima. Maximizar la varianza explicada resulta equivalente a minimizar la varianza del error. El enfoque clásico para la resolución de este problema consiste la diagonalización de la matriz de varianzas-covarianzas. Las componentes principales son los autovectores de módulo unidad asociados a los mayores autovalores, que coinciden con la varianza de esa componente principal. Dichos autovalores son ortogonales. Trasladar esta idea a las redes en principio no parece nada sencillo. Diagonalizar matrices, hallar valores y vectores propios,... no parece que esté en la naturaleza de las redes. Se hace necesario buscar otra perspectiva, para reenfocar estos objetivos; en este caso resulta de interés el análisis de las componentes principales desde un punto de vista geométrico.

La primera componente principal cumple que la distancia ortogonal media de la muestra a la recta determinada por esta componente es mínima. Así mismo, la distancia media ortogonal de la muestra al plano determinado por las dos primeras componentes principales es mínima. La propiedad análoga se cumple para los siguientes subespacios afines determinados por la para las sucesivas componentes principales. Entonces la minimización de distancias será la clave para la búsqueda de las componentes principales, que serán combinaciones de las variables de entrada, así que el diseño de la red será similar al de la regresión lineal, pero sin sesgo.

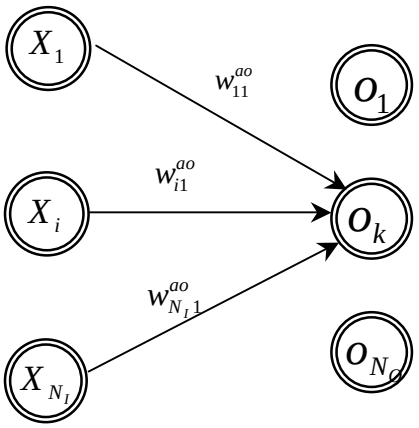

Figura 1.28. Red Neuronal para el Análisis de Componentes Principales

Inicialmente la función de error sería la distancia ortogonal media entre el conjunto de entrenamiento y el subespacio afín determinado por las estimaciones que hacemos de las N_O componentes principales. Considerado así sólo se sabrá que las estimaciones determinan el mismo subespacio que las N_O primeras componentes principales, sin identificarlas en realidad.

Luego, lo más interesante en este problema, es determinar la función que deseamos minimizar. Se desea que O_k corresponda a la k-ésima componente principal, pero con esa primera función de error nada asegura que la primera salida estime la primera componente principal ni mucho menos. Sólo se tienen vectores que dan lugar al mismo espacio afín que las componentes principales. Con los multiplicadores de Lagrange se puede conseguir la ortogonalidad dos a dos de los vectores y que tengan módulo unitario, pero las variables siguen sin estar bien identificadas y ni siquiera se sabe si en realidad se están calculando las componentes principales, u otra base ortonormal de ese mismo espacio. Para evitar estos problemas es necesario añadir algunos términos a la función de error. En primer lugar, la distancia ortogonal media del conjunto de entrenamiento a la recta determinada por O_1. De este modo se tiene la certeza de que sea la primera componente principal. En segundo lugar la distancia ortogonal media de la muestra al plano formado por O_1 y O_2. Así, y como son ortogonales y unitarias, O_2 resulta ser la segunda componente principal. Este proceso se extiende de modo recurrente para cada subespacio hasta llegar a tener la función de error completa, que incluye las distancias a cada subespacio afín y los multiplicadores de Lagrange.

1.7 Aproximadores Universales

En las secciones anteriores se estudiaban las redes como aproximadores bien una variable respuesta, bien de una función de densidad (Park y Sandberg, 1991; Scarselli y Tsoi, 1998; Vapnik y Lerner, 1963; Devroye *et al.*, 1996.). En realidad se estaba considerando un espacio de funciones amplio, con ciertas condiciones de regularidad, en el que se tenía la suposición de que se encontraba, bien la función que nos proporciona una relación de regresión, bien la función de densidad de una variable aleatoria.

Los aproximadores universales son un grupo de funciones que constituyen una base de un espacio de funciones, y que, por lo tanto, son capaces de rescribir en términos de una combinación infinita de funciones de la base, cualquier función del espacio. Es esa infinitud lo que hace imposible en muchos casos obtener exactamente la función deseada. Es necesario pues, limitar el número de elementos de la base, S, que nos proporcionarán una aproximación de la función objetivo. La bondad de la aproximación dependerá del valor que tome S. Si S fuese pequeño, el estimador estaría suavizado, aumentando así su sesgo y disminuyendo su varianza. No resulta suficiente con sugerir que el número ha de ser grande, sino que su valor dependerá del tamaño de la muestra de la que se disponga para estimar los parámetros de la combinación lineal, lo que en redes sería el conjunto de entrenamiento. Si el número de funciones y de datos está próximo, esto es, si el número de datos estuviese próximo al número de parámetros a estimar, se produciría un problema de interpolación, que en redes se denomina sobreaprendizaje, de modo que el estimador no sería adecuado. Para ilustrar esta idea a continuación se muestran algunos ejemplos de aproximadores universales.

1.7.1 Estimadores de Desarrollos Ortogonales

Si consideramos el espacio de las funciones (1.115), estamos considerando un espacio Hilbert, y se puede construir una base ortonormal $\{\Psi_i\}$ infinita respecto a la función de pesos ω (Rudin, 1976; Boyce y DiPrima, 2005).

$$L^2(\omega) = \left\{ g : \Re \to \Re / \int g(x)^2 \omega(x) dx < \infty \right\} \quad (1.115)$$

Al ser un conjunto generador se tiene que cualquier función del conjunto puede ser escrita como una combinación lineal de los elementos de la base.

$$\exists \{c_i\} / \ f(x) = \sum_{i=1}^{\infty} c_i \Psi_i(x), \forall f \in L^2(\omega) \quad (1.116)$$

Se puede aplicar tanto a funciones de regresión como a densidades. Los parámetros c_i resultan ser una esperanza

$$c_i =< f, \Psi_i >_{\omega} = E\left[\Psi_i(x) \cdot \omega(x)\right] \quad (1.117)$$

Por lo tanto estos valores pueden ser estimados sin más que sustituir la media teórica por la muestral. El proceso se detendrá en algún término *S*

$$\hat{f}(x) = \sum_{i=1}^{S} \hat{c}_i \Psi_i(x) \quad (1.118)$$

Esta expresión generaliza el desarrollo en series de Fourier (Spivak, 1980; Strichartz, 1995; Askey y Haimo, 1996.)

El parámetro *S* cumple la función que desempeñaron en su momento los parámetros *h*, y *k*. Este estimador es, por construcción, aquel que minimiza el error cuadrático medio.

La traslación a redes neuronales se muestra en la Figura 1.28. Las capas ocultas se encuentran íntimamente relacionadas con las distintas funciones de la base.

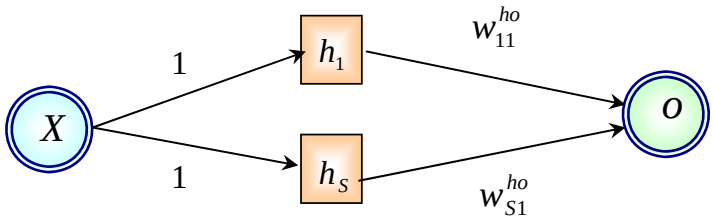

Figura 1.28. Red Neuronal para la Estimación en Desarrollos Ortogonales

Las siguientes ecuaciones muestran el funcionamiento de la red

$$h_i = \Psi_i(x), \ para \ i = 1, \ldots S \quad (1.119)$$

$$o = \sum_{i=1}^{S} \omega_{i1}^{ho} h_i = \sum_{i=1}^{S} \omega_{i1}^{ho} \Psi_i(x), \ para \ i = 1, \ldots S \quad (1.120)$$

En este caso el parámetro de suavizado es el número de nodos de la capa funcional oculta. Se puede apreciar que esta capa no aumenta el número de pesos, pues sus conexiones de entrada son constantes. No es conveniente hacer que los pesos de la capa de entrada a la oculta sean variables, pues esto conlleva la pérdida de la unicidad del modelo.

Un ejemplo sería considerar el espacio [0,1],y la base formada por,

$$\Psi_k(x) = \exp(2\pi ikx),\ para\ k = \pm 1, \pm 2 \ldots \tag{1.121}$$

Sería el desarrollo de una función como *Serie de Fourier*; como los coeficientes de Fourier están definidos para subíndices positivos y negativos, las sumas podrían extenderse en ambas direcciones (truncadas en $\pm S$). En ocasiones los modelos se combinan. Por ejemplo, es posible emplear un desarrollo en series de Taylor truncado como si fuese un núcleo, y luego emplearlo en los esquemas de regresión o en los de estimación de la densidad, como cualquier otra función tipo núcleo. (Ripley,1996)

Asociados a los desarrollos en series de Taylor surgen los núcleos de *Dirichlet* (1.122) y de *Fejér* (1.123) (Silverman, 1973; Spivak, 1988; Apostol, 1991)

$$K(x) = \frac{\operatorname{sen}\left[(2S+1)x\right]}{\operatorname{sen}(\pi x)} \tag{1.122}$$

$$K(x) = \frac{1}{S+1} \cdot \frac{\operatorname{sen}\left[(2S+1)x\right]}{\operatorname{sen}(\pi x)} \tag{1.123}$$

En los modelos tipo núcleo la estimación de la variable respuesta (que puede ser 1 ó 0 en la estimación de la densidad) se obtiene como combinación lineal de los valores en todos los centros o en los *k* más próximos, con los coeficientes dependientes de las distancias en virtud de una función tipo núcleo. Si se considera una función *f* se obtiene la expresión:.

$$Y = \sum_{i=1}^{N} f\left(d_i^X\right) \cdot Y_i^X + \varepsilon_X \tag{1.124}$$

Siendo *N* bien el número total de nodos (N_H), bien los más cercanos (*knn*).

Podría considerarse para *f* cualquier función con máximo en cero. Se podría aproximar a través de cualquier familia de aproximadores universales, por ejemplo, por desarrollos ortogonales. Es posible asimismo incorporar a esa función un parámetro ventana (o varios). La estimación de *f* funcionará como un núcleo (de *Dirichlet*, de *Fejér,* ...). También sería una opción la utilización de cualquier base, no sólo las ortogonales, pero estas bases llevan asociados problemas de dependencia.

1.7.2 Funciones Sigmoideas

Teorema1. Si se consideran el conjunto de las funciones $\boldsymbol{f \in C(I_{N_I})}$ con $\boldsymbol{I_{N_I} = [0,1]^{N_I}}$ y $\boldsymbol{\sigma}$ una función continua y sigmoidea, i.e. función real de variable real, con $\lim_{x \to +\infty} \sigma(x) = 1$, y $\lim_{x \to -\infty} \sigma(x) = 0$, se tiene que dado un $\boldsymbol{\varepsilon > 0}$,existe una suma finita de la forma

$$\hat{f}(\vec{x}) = \hat{f}\left(x_1, \ldots, x_{N_I}\right) = \sum_{j=1}^{N_H} \omega_{j1}^{ho} \cdot \sigma\left(\sum_{i=1}^{N_I} \omega_{ij}^{ah} x_i + \omega_{0j}^{ah}\right),\ con\ \ N_H = N_H(\varepsilon) \tag{1.125}$$

De tal modo que $\left| f(\vec{x}) - \hat{f}(\vec{x}) \right| < \varepsilon, \forall \vec{x} \in I_{N_I}$

Luego, empleando una red neuronal con función de activación sigmoidea en la capa oculta, e identidad en la de salida, es posible aproximar cuanto se desee cualquier función suave, sin más que considerar una cantidad suficiente de nodos en la capa oculta. (Lang, 2005; Lugosi y Zeger, 1995). El número de nodos de esta capa oculta desempeña el mismo papel que el parámetro de suavizado en la regresión tipo núcleo. Este teorema constituye la base teórica del funcionamiento de los perceptrones son una única capa oculta. Entre las funciones sigmoideas se encuentran algunas tan conocidas como la *logística* y la *tangente hiperbólica*.

La red asociada será un perceptrón con una capa oculta, que se corresponde con la Figura 1.6, salvo que no se considera el término de sesgo entre la capa oculta y la de salida. El funcionamiento de una red de este tipo responde a las expresiones siguientes.

$$h_j = \sigma\left(\sum_{i=1}^{N_i} \omega_{ij}^{ah} \cdot X_i + \omega_{0j}^{ah}\right) \quad para\ \ j = 1,\cdots,N_H \tag{1.126}$$

$$o = \sum_{j=1}^{N_H} \omega_{j1}^{ho} \cdot h_j + \omega_{j1}^{ho} \tag{1.127}$$

1.7.3 Funciones Tipo Núcleo

Teorema2. Sea K una función es continua, acotada y con integral finita no nula, la clase (1.128) es densa en L_p ($\forall$ p$\in$[1,∞)) y puede aproximar uniformemente funciones continuas en compactos.

$$y = \omega_{01}^{ho} + \sum_{i=1}^{N_H} \omega_{i1}^{ho} K\left(\frac{\left\|\vec{x} - \vec{W}_i^{ah}\right\|}{\omega_0^{ah}}\right) \tag{1.128}$$

Para toda *f* existirá un conjunto de centros $\left\{\vec{W}_i\right\}_{i=1}^{N_H} = \left\{\vec{W}_i^{ah}, \vec{W}_i^{ho}\right\}_{i=1}^{N_H}$ y $\omega_0^{ah} > 0$ de modo que (1.128) aproxima a *f*, empleando la norma adecuada.

Este resultado constituye la base teórica del funcionamiento de las redes de base radial en general, de modo se obtienen aproximaciones uniformes de funciones continuas, sobre soportes compactos.

1.8 Redes Probabilísticas

Las redes probabilísticas constituyen, junto con las redes neuronales, una de las estructuras de aprendizaje más extendidas en la actualidad. Estas redes se enmarcan dentro de los denominados *sistemas expertos.* El uso de estas redes se ha extendido fundamentalmente a problemas de clasificación.

Son, como se acaba de señalar, sistemas expertos. Una de las definiciones de sistema experto puede definirse como un sistema informático que simula el comportamiento que tendría el experto de un cierto campo. Existen dos tipos de sistemas expertos, aquellos que tratan con problemas determinísticos, y aquellos que se enfrentan a problemas de naturaleza estocástica.

Las redes probabilísticas se enmarcan dentro de los sistemas *probabilísticos*, pues se enfrentan a problemas con elementos de incertidumbre relevantes. En particular son modelos

probabilísticos *definidos gráficamente*, esto es, pueden ser representados a través de un grafo, por lo que en muchas ocasiones se identifican redes y grafos.

Las redes probabilísticas pueden definirse pues como grafos que conllevan asociados funciones que relacionan los nodos conectados, que pueden ser bien probabilidades condicionadas o bien funciones potenciales dependiendo del tipo de grafo y si son dirigidos o no. Entre los diferentes tipos de las redes probabilísticas destacan las redes bayesianas y las redes de Markov, que se diferencian entre otras cosas por el tipo de grafo subyacente (Castillo *et al.*, 1996).

Las redes probabilísticas tienen dos objetivos fundamentales. En primer lugar estimar, a partir de un conjunto de datos disponibles (información), distribuciones de probabilidad condicionadas y/o estructuras de dependencia, a través de algún método de *aprendizaje*; en segundo lugar generar nuevos conocimientos a través de las denominadas técnicas de *propagación* de la evidencia.

Con respecto al aprendizaje, hay dos tipos fundamentales de métodos: los de tipo *estructural* y los de tipo *paramétrico*.

Los primeros se centran en la búsqueda e identificación de la estructura de las relaciones de dependencia y correlación a través de la estructura del grafo que define la red; por su parte los de tipo *paramétrico* están ligados a la estimación de los valores de los parámetros asociados a los nodos *y* aristas del grafo que representa la red.

Ambas búsquedas, las de la estructura y los parámetros han de ir de la mano, pues no se puede seleccionar una estructura hasta comprobar que tras la estimación de parámetros, esta funciona adecuadamente.

Es necesario pues definir medidas de *calidad* de la red que combinen la bondad de las estimaciones y su coherencia con la información disponible, con la simplicidad del esquema y otras medidas asociadas a las características estructurales del grafo, así como la opinión del experto. Esta elección de la medida de calidad es pues, más complicada e involucra más elementos que la habitual selección de una función de error de otros modelos de aprendizaje.

Existen multitud de medidas propuestas como la de Cooper-Herskovits (1992), la de Geiger y Heckerman (1995) y la de Medida de Calidad Bayesiana usual (Castillo *et al.*, 1996)

Cabe señalar que varias estructura de grafos pueden representar dirigidos pueden representar las mismas estructuras de independencia y/o las mismas distribuciones conjuntas para el conjunto de sus variables.

Algunos de los algoritmos de aprendizaje para redes bayesianas más conocidos son el Algoritmo K2 (Cooper y Herskovits, 1992) y el Algoritmo B (Buntine, 1991) para datos completos, esto es, cuando todas las observaciones del conjunto de información constan de los valores para todas y cada una de las variables; si el conjunto de datos no es completo destacan el algoritmo EM (Dempster *et al.*, 1977) y el muestreo de Gibbs (Gelfand y Smith, 1990).

Una vez construida la red probabilística con mayor medida de calidad a partir de los datos disponibles, se pasa al siguiente objetivo de la red. La obtención de nuevos conocimientos a partir de la estructura de representada en la red y de nueva información disponible. Cuanod no se dispone de nueva información, o *evidencia* se calculan las probabilidades marginales de los

nodos, que proporcional la información a priori sobre los distintos valores que pueden tomar las variables. En caso de que sí se disponga de nueva información con el fin de calcular las distribuciones de probabilidad condicionadas a la evidencia conocida, se emplean los llamados métodos de propagación de la evidencia, o simplemente de propagación. Dichos algoritmos se pueden clasificar en tres grandes grupos: los *exactos*, los *aproximados* y los *simbólicos*.

Los algoritmos *exactos* son aquellos que permiten calcular las probabilidades asociadas a os nodos del grafo de modo exacto (salvo errores de redondeo asociados a limitaciones computacionales). Estos algoritmos están determinados por el tipo de grafo asociado a la red peorbabilística.

Los algoritmos *aproximados* por el contrario no calculan las probabilidades exactas sino que, empleanso técnicas de simulación obtienen valores aproximados. Se emplean cuando los algoritmos exactos son computacinalmente muy costosos o incluso no es posible aplicarlos. Algunos de los métodos exactos sufren de un problema de explosión combinatoria.

Los algoritmos de propagación simbólica tienen la propiedad de poder operar no sólo con parámetros numéricos sino también con parámetros simbólicos, lo que permite obtener las probabilidades en función de los parámetros, esto es, en expresión simbólica.

Los métodos de propagación numéricos, tanto exactos como aproximados, requieren que los parámetros tengan asignados valores numéricos fijos. Puede que esto no sea posible conocer el valor, o bien se conozca sólo el intervalo de pertenencia para alguno de los párametros en lugar de sus valores exactos. En tales casos es adecuado emplear métodos simbólicos, que son capaces de tratar los parámetros mismos, sin necesidad de asignarles valores.

Estas redes probabilísticas se aplican con éxito a muy diversos problemas, desde la ingeniería, con la modelización de estructuras físicas, a la medicina con aplicaciones al diagnóstico, pasando por la economía, la planificación de problemas,…

1.9 Resumen

Se ha podido estudiar en este capítulo que muchos métodos estadísticos, desde los más clásicos a los más vanguardistas, pueden ser reescritos desde la perspectiva de las redes neuronales. En algunos casos, éstas plantean variantes que pueden ser de interés, mientras que en otros casos no aportan novedades sobre los modelos ya conocidos. Las redes son herramientas amplias de trabajo, que no requieren ningún conocimiento sobre los datos, y es por ello que se han hecho tan populares. Pero se siguen necesitando estudios más minuciosos sobre su eficiencia y consistencia, y sobre los problemas derivados de los métodos de entrenamiento empleados. No debemos abandonar alegremente los planteamientos teóricos de las redes neuronales. Bien es cierto que no requieren ningún tipo de explicación del modelo, que puede incluso parecer incomprensible (ventanas negativas,...), pero esto es al tiempo una ventaja y un inconveniente. Muchos de los modelos que generan se resisten a interpretaciones de los problemas reales a los que responden. La combinación de estudios estadísticos más asentados, con el uso de redes neuronales será lo que mejores resultados nos proporcione. En los siguientes capítulos se abordará la aplicación de redes neuronales a la resolución de problemas de diversa índole en ámbitos medioambientales e industriales.

1.10 Bibliografía

Aluja Banet, T. y Morineau, A. (1999) Aprender de los datos : el análisis de componentes principales : una aproximación desde el data mining. Universidad de Barcelona.

Apóstol, T. M. (1991) Análisis Matemático. (2 Ed.) Reverté.

Artiaga, R., Cao, R., Naya, S. (2003) Local polynomial estimation of TGA derivatives using logistic regression for pilot bandwidth selection ". Thermochimica Acta, V.406, pp. 177-183.

Askey, R. y Haimo, D. T. (1996) Similarities between Fourier and Power Series. Amer. Math. Monthly V.103, pp. 297-304.

Berger, J. (1985) Statistical Decision Theory and Bayesian Analysis. Springer Verlag.

Bertsekas, D.P. (1999) Nonlinear Programming.(2 Ed.) Athena Scientific.

Bishop, C. (1995) Neural Networks for Pattern Recognition, Oxford: Clarendon Press.

Boyce, W.E. y DiPrima, R.C. (2005) Elementary Differential Equations and Boundary Value Problems, Eighth edition. John Wiley & Sons, Inc.

Buhmann, M.D. (2003). Radial Basis Functions: Theory and Implementations. Cambridge University.

Buntine, W. (1991) Theory Refinement on Bayesian Networks. In Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence. Morgan Kaufmann Publishers, San Mateo, CA, pp. 52–60.

Cachero, M.L. (1996) Fundamentos y métodos de estadística. Pirámide.

Canavos, G.C. (2003) Probabilidad y Estadística. Aplicaciones y Métodos. McGraw-Hill.

Cao, R., Cuevas, A. y González Manteiga, W. (1994) A comparative study of several smoothing methods in density estimation. Computational Statistics & Data Analysis, V.17(2), pp. 153-176.

Castillo, E. Gutiérrez, J.M y Hadi, A.S. (1996) Sistemas Expertos y Modelos de Redes Probabilísticas. Colección de Monografías. Real Academia de Ingeniería.

Cooper, G.F. y Herskovits, E. (1992), A Bayesian Method for the Induction of Probabilistic Networks from Data. Machine Learning, V.9, pp. 309–347.

Cover, T. M. y Hart, P. E. (1967) Nearest neighbor pattern classification. IEEE Transactions on Information Theory, V.13, pp. 21-27.

Cuadras, C.M. (1981) Métodos de Análisis Multivariante. Eunibar.

Chakraborty, K., Mehrotra, K. et al (1992): Forecasting the Behavior of multivariate time series using neural networks, Neural Networks, V.5, pp. 961-970.

Dasarathy, B.V. (1991). Nearest Neighbour (NN) Norms: NN Pattern Recognition Techniques. IEEE Computer Society Press.

Delecroix, M., Härdle, W. y Hristache, M. (2003) Efficient estimation in conditional single-index regression, Journal of Multivariate Analysis, V.86(2), pp. 213-226.

Dempster, A., Laird, N., y Rubin, D. (1977) Maximum Likelihood from Incomplete Data Via the EM Algorithm. Journal of the Royal Statistical Society, B, V.39, pp. 1–38.

Devroye, L. Györfi, L. y Lugosi, G. (1996) A Probabilistic Theory of Pattern Recognition. Springer-Verlag.

Dobson, A.J. (1990) An Introduction to Generalized Linear Models. Chapman & Hall.

Duda, R.O., Hart, P.E., Stork, D.G. (2001) Pattern Classification. Wiley-Interscience.

Everitt, B. S., Landau, S. y Leese, M. (2001) Cluster analysis. (4 Ed.) Oxford University Press.

Fan, J. y Gijbels, I. (1992) Variable bandwidth and local linear regression smoothers. Annals of Statistics, V.20, pp. 2008–2036.

Fox, J. (2008) Applied Regression Analysis and Generalized Linear Models. Sage Publications.

Franke, J. y Neumann, M.H. (1998): Bootstrapping neural networks, Report in Wirstschaftsmathematik 38/1998, FB Mathematik, University of Kaiserslautern.

Friedman, J. H. y Stuetzle, W. (1981) Projection Pursuit Regression. *Journal of the American Statistical Association*, V.76(376), pp. 817-823.

Friedman, J. H. y Tukey, J. W. (1974). A Projection Pursuit Algorithm for Exploratory Data Analysis. IEEE Transactions on Computers C-23, V.9, pp. 881–890.

Fukunaga, K. (1990) Introduction to statistical pattern recognition. (2 Ed.) Academic Press.

Gasser, T. y Müller, H. G. (1984) Estimating regression functions and their derivatives by the kernel method, Scandinavian Journal of Statistics, V. 11, pp. 171-185.

Gasser, T. Müller, H.G. y Mammitzsch, V. (1985) Kernels for Nonparametric Curve Estimation Journal of the Royal Statistical Society Series B, V.47(2), pp. 238-252.

Gasser, T., Kneip, A., Köhler, W. (1991) A flexible and fast method for automatic smoothing. Journal of the American Statistical Association, V.86, pp. 643- 652.

Geiger, D. y Heckerman, D. (1995) A Characterization of the Dirichlet Distribution with Application to Learning Bayesian Networks. In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. Morgan Kaufmann Publishers, San Francisco, CA, pp. 196–207.

Gelfand, A.E. y Smith, A.F. (1990) Sampling-Based Approaches to Calculating Marginal Densities. Journal of the American Statistical Association, V.85, pp. 398–409.

Golberg, M.A. y Cho, H.A. (2004) Introduction to Regresion Analysis. WIT Press.

Hall, P., Sheather, S. J., Jones, M. C. y Marron, J. S. (1991) On optimal data-based bandwidth selection in kernel density estimation. Biometrika, V.78, pp. 263-269.

Hair, J.F., Anderson, R.E., Tatham, R.L., Black, W.C. (1999) Análisis Multivariable. (5 Ed.) Prentice – Hall.

Härdle, W. (1990) Applied Nonparametric Regression, Econometric society monographs, Cambridge University Press.

Härdle, W. Hall, P. y Marron, J. S. (1992) Regression smoothing parameters that are not far from their optimum. Journal of the American Statistical Association, V.87, pp. 227-233.

Härdle, W. y Marron, J. S. (1985a) Optimal bandwidth selection in nonparametric regression function estimation. The Annals of Statistics, V.13, pp. 1465-1481.

Härdle, W. y Marron, J. S. (1985b) Bandwidth choice in nonparametric regression function estimation. Statistics and Decisions, Sup. 2, pp. 173-177.

Härdle, W. and Stoker, T. M. (1989) Investigating smooth multiple regression by the method of average derivatives. Journal of the American Statistical Association. V.84, pp. 986-995.

Harman,H.H. (1976) Análisis Factorial Moderno. Saltés.

Hart, P.E. (1968). The condensed nearest neighbour rule. IEEE Transactions on Information Theory, IT-14, pp. 515-516.

Hartman, E. Keeler, J. D. y Kowalski, J. M. (1990) Layered neural networks with gaussian hidden units as universal approximations. Neural Computation,V.2(2), pp. 210–215.

Hastie, T. y Tibshirani, R. (1990) Generalized Additive Models. Chapman and Hall.

Hastie, T., Tibshirani, R. y Friedman, J. (2009) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. (2 Ed.) Springer.

Haykin, S. (1999) Neural Networks. A Comprehensive Foundation. (2 Ed.) Pretince Hall.

Haykin, S. (2009) Neural Networks and Learning Machines (3 Ed.) Prentice Hall.

Hilera González, J.R. y Martínez Hernando, V.J. (1995) Pattern Recognition Using Neural Networks, Cambridge University Press.

Ichimura, (1993) Semiparametric least squares (SLS) and weighted SLS estimation of single-index models. Journal of Econometrics, V.58, pp. 71-120.

Jones, M.C., Marron, J.S., Sheather, S.J. (1996) A Brief Survey of Bandwidth Selection for Density Estimation. Journal of the American Statistical Association, V.91, pp. 91: 401-407.

Kaufman, L. y Rousseeuw, P.J. (1990) Finding groups in data : an introduction to cluster analysis. Wiley.

Lachenbruch, P.A.(1975) Discriminant Analysis. Hafner Press.

Lang, B (2005) Monotonic Multi-layer Perceptron Networks as Universal Approximators. Springer.

Looney, C.G. (1997) Pattern Recognition and Neural Networks, Cambridge University Press.

Lugosi G. y Zeger, K. (1995) Nonparametric Estimation via Empirical Risk Minimization. IEEE Transactions on Information Theory, V.41(3), pp. 677-687.

McCullagh, P. and Nelder, J.A. (1989) Generalized Linear Models. (2 Ed.) Chapman & Hall.

Montgomery, D.C., Peck, E.A. y Vining, G.G. (2005) Introducción al Análisis de Regresión Lineal. Compañía Editorial Continental.

Müller, H.G. y Stadtmuller, U. (1987) Variable Bandwidth Kernel Estimators of Regression Curves. Annals of Statistics, V.15(1), pp. 182-201.

Nadaraya, E.A. (1964) On estimating regression, Theory of Probability and its Applications, V.9, pp. 141-142.

Park, J., Sandberg, I. (1991) Universal approximation using radialbasis function networks. Neural Computation, V.3, pp. 246-257.

Peña, D. (2002) Análisis de datos multivariantes. McGraw Hill.

Peña, D. (2002) Regresión y Diseño de Experimentos. Alianza Editorial.

Powell, M. J. D. (1987) Algorithms for the Approximation of Functions and Data. Clarendon Press.

Priestley, M.B. y Chao, M.T. (1972) Non-parametric function fitting, Journal of the Royal Statistical Society, Series B, V.34, pp. 385-392.

Ripley, B.B. (1996) Pattern Recognition using Neural Networks. Cambridge University Press.

Rosenblatt, F. (1958) The perceptron: A probabilistic model for information storage and organization in the brain. Psycological Review, V.65, pp. 386-408.

Rosenblatt, F. (1962) Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books.

Rudin, W. (1976) Principles of mathematical analysis. (3 Ed.) McGraw-Hill, Inc.

Rumelhart, D.E., Hinton, G.E., y Williams, R.J. (1986) Learning Representations by Back-propagating errors. Nature, V.323, pp. 533-536.

Ryan, T.P. (1997) Modern Regression Methods. Wiley.

Sarle, W.S. (1994) Neural networks and statistical models, In Proceedings of the 19th Annual SAS Users Group International Conference.

Scarselli, F. y Tsoi, A.C. (1998) Universal Approximation Using Feedforward Neural Networks: A Survey of Some Existing Methods, and Some New Results. Neural Networks, V.11(1), 12, pp. 15-37.

Scofier, B. y Pagès, J. (1992) Análisis factoriales simples y múltiples. Objetivos, métodos e interpretación. Servicio Editorial de la Universidad del País Vasco, Bilbao.

Scott, D. W. (1992) Multivariate Density Estimation: Theory, Practice, and Visualization. John Wiley and Sons, Inc.

Scott, D.W. (2008) Multivariate Density Estimation. John Wiley and Sons, Inc.

Seber, G.A.F. y Wild, C.J. (2003) Nonlinear Regression. Wiley.

Sheather, S. J. y Jones, M. C. (1991) A reliable data-based bandwidth selection method for kernel density estimation. Journal of the Royal Statistical Society, Series B, V.53, pp. 683–690.

Silverman, R. A. (1973) Complex Analysis with Applications, Dover Publications.

Silverman, B.W. (1986) Density Estimation for Statistics and Data Analysis. Chapman and Hall.

Simonoff, J. S. (1996). Smoothing Methods in Statistics. Springer.

Spivak, M. (1980) Calculus. Reverté.

Spivak, M. (1988) Cálculo Infinitesimal. Reverté.

Stoker, T. M. (1986). Consistent estimation of scaled coefficients. Econometrica, V.54, pp. 1461-1481.

Stone, C.J. (1977) Consistent non parametric regression. Annals of Statistics, V.5, pp. 595-645.

Strichartz, R. (1995) A Guide to Distribution Theory and Fourier Transforms. gCRC Press.

Tang, H., Tan, K.C., Yi, Z. (2007) Neural Networks: Computational Models and Applications. Series: Studies in Computational Intelligence, V.53. Springer.

Tou, J.T. y González, R.C. (1974) Pattern recognition principles. Addison-Wesley.

Vapnik, V. y Lerner, A. (1963) Pattern recognition using generalized portrait method. Automation and Remote Control, V.24, pp.774–780.

Wand, M.P. y Jones, M.C. (1995) Kernel Smoothing, Chapman & Hall.

Wasserman, L. (2005) All of Nonparametric Statistics. Springer.

Watson, G.S. (1964) Smooth regression analysis, Sankhya, Series A, V.26, pp. 359-372.

Widrow, B. y Hoff, M. (1960) Adaptive switching circuits, In Western Electronic Show and Convention, Convention Record, V.4, pp. 96-104. Institute of Radio Engineers.

Wood, S.N. (2006) Generalized Additive Models. An Introduction. Chapman and Hall.

Wu, T.J., Chena, C.F. y Chen, H.Y. (2007) A variable bandwidth selector in multivariate kernel density estimation. Statistics and Probability Letters. V. 77(4), pp. 462-467.

CAPÍTULO 2. MODELIZACIÓN DE VARIABLES CONTINUAS CON REDES NEURONALES.

RESUMEN

La potencia predictora de las redes neuronales se ha proyectado a muy diversos ámbitos. En este segundo capítulo se mostrará como una red neuronal puede ser empleada para la predicción de variables consideradas a lo largo del tiempo. Se ha considerado la aplicación a un caso real, la modelización de comportamiento hidrográfico de la cuenca de un río. En particular se ha modelizado el comportamiento de las aportaciones, recibidas por el río, a diferentes horizontes temporales. Para un horizonte a largo plazo, mensual, se verá la eficiencia de la modelización Box-Jenkins, tanto para modelizar las aportaciones medias de la cuenca como las lluvias medias registradas. Para un horizonte a corto plazo, en particular diario, se han empleado redes neuronales, y se han comparado con los resultados de los modelos Box-Jenkins. La capacidad de las redes neuronales de modelar una relación compleja de lluvias-escorrentía ha sido constatada. Aunque el funcionamiento de la red neuronal no fue satisfactorio para descubrir algunos picos, los resultados son prometedores.

Parte de los resultados que se detallan en este capítulo están recogidos en Castellano-Méndez *et al.*, 2004.

2.1 Introducción

Las redes neuronales artificiales (RNA) son, como se mostró en el capítulo anterior, una herramienta muy útil para el análisis de datos. Su uso se hace de particular interés cuando desconocemos por completo la estructura del fenómeno subyacente a los datos que tratamos de analizar, el campo de la estadística no paramétrica. Resultan, pues, útiles cuando se intentan explicar problemas físicos complejos cuyo estudio requiere determinar los valores de muchos parámetros, no siempre fáciles de estimar a partir de las observaciones reales. En estos casos, se pueden obtener redes neuronales que suplan o complementen a los modelos físicos que reproducir y explicar un determinado fenómeno. En general se pueden construir y diseñar redes que simulen el comportamiento del fenómeno físico en cuestión, de modo que los parámetros desconocidos se estimen durante el proceso de entrenamiento. Alternativamente se puede considerar un modelo general, y en base a las propiedades de aproximador universal que presentan las redes neuronales con determinada estructura, tal y como se comentó en el capitulo 1, (Cybenko, 1989; Hornik *et al.*, 1989; Park y Sandberg, 1991, Castro *et al.*, 2000), encontrar la red que reproduzca el fenómeno que se desea estudiar.

2.1.1 Introducción al Problema Hidrológico

En esta era tecnológica, uno de los principales problemas de la sociedad es obtiene un suministro adecuado y fiable de energía. El agua del río es una fuente de energía con la ventaja de ser tanto renovable como no contaminante, pero esto sufre la desventaja de que su suministro está a merced de la naturaleza: ni la cantidad del agua que el río recibe, ni momento, ni intensidad con que ésta llega, puede ser controlada. La explotación óptima hidroeléctrica de un río requiere un conocimiento de la futura disponibilidad de recursos, en este caso la cantidad del agua que estará disponible en el futuro, planificar la cantidad de energía que se va a generar, y el instante adecuado para hacerlo.

Para facilitar el pronóstico de los recursos hidrológicos, se han desarrollado muchas técnicas diferentes durante los últimos años, desde modelos simples, como el modelo de Tanque (Sugawara, M., 1974, 1979, 1995) o el método de primitivas racionales, hasta modelos sofisticados, basado en modelos físicos distribuidos, por ecuaciones parciales diferenciales, como el SHE (Systèm Hydrologique Europeèn) (Abbot *et al.*, 1986), o su evolución el MIKE SHE (Refsgaard y Storm, 1995). Otros interesantes modelos físicamente inspirados son el modelo conceptual de Xinanjiang, (Zhao *et al.*, 1980), satisfactoriamente usado en China, y el PRMS (Precipitation-Runoff Modelling System), (Leavesley *et al.*, 1983). Los modelos de lluvia-escorrentía pueden ser clasificados dependiendo del grado de representación de los procesos subyacentes físicos, como modelos de caja negra, modelos conceptuales y distribuidos y modelos físicamente basados.

Los modelos conceptuales y modelos físicamente basados se diseñan tratando de simular matemáticamente los mecanismos físicos que determinan el ciclo hidrológico, y suelen implicar leyes físicas de transferencia de agua , y parámetros asociados con las características de la cuenca. La calibración de tales modelos es compleja y requiere de un conocimiento profundo de la cuenca estudiada (Sorooshian y Gupta, 1995). Una revisión de técnicas relevantes matemáticas puede ser encontrada en el artículo de Singh y Woolhiser (2002).

Los métodos de caja negra son métodos a basados en los datos, que se han convertido en métodos muy populares pues al emplearnos es posible evitar el problema de entender la estructura inherentes a los procesos que tienen lugar en el sistema que está siendo modelado. El análisis de serie de tiempo, y redes neuronales artificiales son métodos de caja negra aplicados satisfactoriamente en pronosticar procesos de lluvia-escorrentía (denominados habitualmente por sus siglas inglesas Rainfall - Runofff, R-R) en diferentes horizontes temporales. Las funciones de transferencia y las red neuronal son los métodos basados en datos más populares. Algunos artículos han abordado objetivamente la comparación entre funciones de transferencia no lineales y métodos de redes neuronales (Lekkas *et al.*, 2001).

Las dificultades a la hora de obtener una interpretación física del modelo de predicción es una característica de los métodos de caja negra. Para las redes sin capa oculta, la fuerza de las relaciones internas puede ser estudiada a partir del análisis directo de los pesos de las conexiones (Tang *et al.*, 1991), pero para redes neuronales de arquitectura más general el estudio de las relaciones entre entradas y salidas se convierte en una tarea imposible (Chakraborty *et al.*, 1992)

Este capítulo se centra en el análisis y la comparación de los modelos Box-Jenkins (Box y Jenkins, 1976) y de las redes neuronales artificiales (RNA) (Rosenblatt, 1958; Rumelhart *et al.*, 1986, 1996), para el análisis de serie de tiempo.

Ambas técnicas han sido satisfactoriamente aplicadas a problemas hidrológicos. Las redes neuronales se usan comúnmente en la práctica (Zelanda *et al.*, 1999; Sajikumar y Thandaveswara, 1999). Incluso cuando los datos hidrológicos sufren de datos faltante, estos métodos pueden ser empleados para estimar los registros hidrológicos faltantes (Khalil *et al*., 2001).

La clase de los modelos autorregresivos de media móvil (ARMA) ha sido el método estadístico más extensamente empleado para modelar series de tiempo, en particular las series de escorrentía (Raman y Sunilkumar, 1995), pero la no linealidad del proceso R-R limita el uso de esta familia de modelos. Algunos autores han comparado los métodos Box-Jenkins con los métodos de RNA (Tang *et al*., 1991; Hsu *et al*., 1995; Abrahert y See, 1998) confirmando, en la mayoría de los casos la confirmación de un mejor funcionamiento de las redes neuronales.

Para la realización de un presupuesto de generación de energía, y por tanto para establecer un presupuesto económico para el próximo año hidrológico, las predicciones de aportaciones o caudales del próximo año se hacen imprescindibles. Es necesario, pues, estimar cuánta lluvia caerá en la cuenca del río el siguiente año, y qué cantidad llegará al río, con el objetivo de decidir el momento y la cantidad que ha de ser generada. En el momento de esta investigación, el precio de electricidad en el mercado español hidroeléctrico varía según el tiempo de día y la estación. Las predicciones a largo plazo son útiles a la hora de estimar si habrá bastante materia prima para generar la electricidad a las horas de precios más altos a lo largo del año. Usando esta información podría ser necesario aplazar la generación para garantizar que la generación eléctrica sea constante, y que el precio obtenido sea el mejor. Se han realizado muchos estudios para estudiar la relación lluvia-escorrentía (R-R) a largo plazo, de modo que los pronósticos mejoran cuando se consideran fenómenos globales como el ENSO (El Niño/Oscilación del Sur) (Uvo *et al*., 2000, Dolling y Varas, 2002)

Por otra parte, es necesario examinar el caudal en la presa cuando se enfoca en problema con una perspectiva a corto plazo. El objetivo principal de la valoración a corto plazo de la escorrentía es evaluar el riesgo de inundaciones, con el fin de tomar medidas para evitar desbordamientos o reducir al mínimo el daño producido. Por lo tanto es también necesario estudiar la evolución de escorrentía obtener predicciones a corto plazo. El horizonte de predicción puede variar desde la predicción semanal (Zelanda *et al.*, 1999), a la predicción diaria (Coubali *et al.*, 2000), o incluso la predicción a una hora (García-Bartual, 2002) y en tiempo real (Deo y Thirumalaiah, 2000). Se dispone de registros diarios de lluvia en cada pluviómetro (l/m^2) y medias de escorrentía (m^3/seg) en cada presa.

El objetivo, en este capítulo, es predecir las aportaciones del río Xallas, localizado en Galicia, en el noroeste de España. La posición geográfica es reflejada en la Figura 2.1. A lo largo del curso del río Xallas se encuentran situadas dos presas: la primera de situada al final de la cabecera, se denomina presa Fervenza e implica una cuenca de 318 km^2; Santa Eugenia es menor que la anterior y se localiza casi en la desembocadura del río. Además en la cuenca se encuentran situados dos azudes, denominados Puente Olveira y Castrelos. Para predecir las

aportaciones del río, se tienen datos de lluvia, registrados por tres pluviómetros situados a lo largo del río, en Fervenza, Puente Olveira, y Santa Eugenia.

El estudio de las aportaciones se puede enfocar de diferentes maneras según el horizonte en el que se deseen hacer las predicciones. Se ha abordado el estudio de las predicciones de las precipitación medias mensuales y escorrentías en las presas de Fervenza y de Santa Eugenia, usando modelos Box-Jenkins, pero sólo se consideró el estudio de las aportaciones diarias en Fervenza. La localización de la presa de Santa Eugenia es tal, que cualquier desbordamiento en este punto simplemente desembocará en el mar, en forma de cascada, sin poner en peligro las localidades cercanas; es por ello que se ha limitado el estudio al comportamiento diario de las aportaciones en la presa de Fervenza.

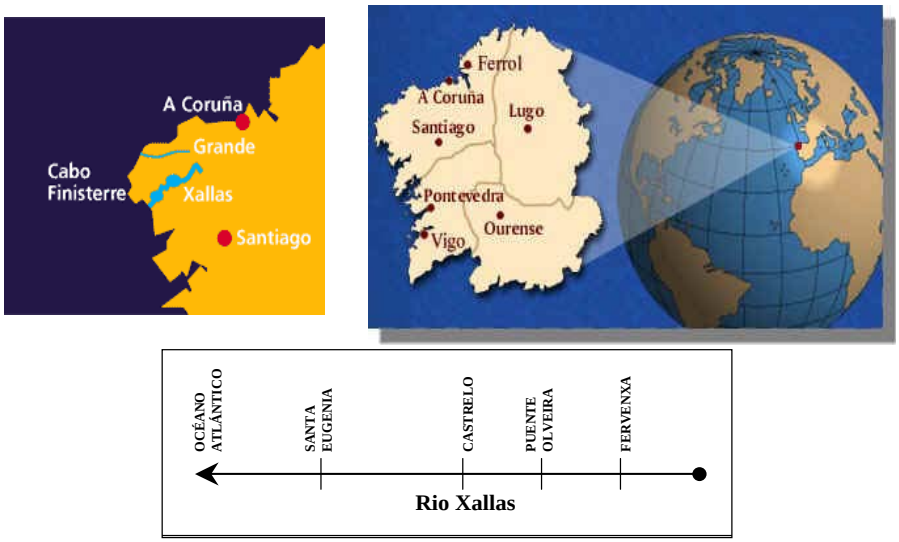

Figura 2.1 Localización del río y su estructura.

El comportamiento mensual difiere considerablemente del que tiene lugar en intervalos de tiempo diarios, y en consecuencia los distintos enfoques pueden proporcionar resultados diferentes para los dos escenarios. El estudio de las aportaciones medias mensuales ha sido abordado a partir de técnicas clásicas de serie de tiempo como los modelos de Box-Jenkins. Sin embargo, el comportamiento de estos modelos está lastrado debido a la no linealidad de los procesos R-R. Así mismo los métodos Box-Jenkins requieren variables estacionarias y normalmente distribuidas, hipótesis que no son necesarias para los métodos de RNA, (Burke, 1991; French *et al.*, 1992); las RNA son capaces de soslayar muchas de las deficiencias que las series de tiempo hidrológicas puede presentar, como discontinuidades y ausencia de datos. Los métodos de redes neuronales no sólo obtienen mejores estimaciones de valores faltantes (Khalil *et al.*, 2001), sino que también pueden ser entrenados sin sufrir tanto estas carencias porque las redes pueden considerar las entradas como variables, sin considerar que están asociados a una serie de tiempo y pueden ser entrenadas usando los datos que no están en un escenario cronológico. Varios estudios han mostrado la capacidad de las redes neuronales de "*aprender*" el proceso Lluvia-Escorrentía (R-R) a partir de conjuntos de datos ambiguos y alterados por ruido. (Nor *et al.*, 2001)

El enfoque de la predicción diaria hace necesario un conocimiento más exhaustivo de la cuenca del río Xallas. Un modelo basado en leyes físicas, que podría dar una explicación satisfactoria

de la relación entre la precipitación y el aumento del flujo o caudal, necesariamente tendría asociados muchos parámetros que tendrían la misión de reflejar los rasgos topológicos, hidrológicos y ecológicos de la cuenca. La compleja tarea de encontrar una modelo estructural adecuado, la cantidad de parámetros y la dificultad de estimar sus valores adecuados así como las complejidades de sus mediciones hizo que se rechazase la consideración del modelo físico, y por el contrario se abordase el uso de redes neuronales artificiales para modelizar este problema. Se emplearán, pues, redes neuronales para la reproducción del proceso R-R, sin considerar las características del fenómeno, pero las RNA también podrían emplearse satisfactoriamente podría calibrar o estimar algunos parámetros del modelo conceptual (Maier y Dandy, 1996)

En la segunda sección de este capítulo se describirán la metodología Box-Jenkins, la regresión dinámica y los modelos diferentes modelos mensuales de predicción de escorrentía adoptados. En la tercera sección, se describirá la arquitectura de las redes neuronales seleccionada para el estudio de datos diarios; no se ahondará en la descripción de las redes, pues ya han sido extensamente descritas en el primer capítulo. En la cuarta sección se presentan los resultados para ambos enfoques del problema, el mensual y el diario. Finalmente, las conclusiones del estudio están contenidas en la sección cinco.

2.1.2 Terminología y Notación

Se ha considerado la predicción de la escorrentía desde dos puntos de vista diferentes, según los horizontes temporales que se han establecido. La notación empleada también refleja el horizonte temporal en el que se encuadre el estudio. Se denota a la aportación o escorrentía media mensual en Fervenza y Santa Eugenia en el instante *t* como por $MMFRO_t$ Y $MMSERO_t$, respectivamente. La escorrentía media diaria en Fervenza y Santa Eugenia ha sido denotada por $DMFRO_t$ Y $DMSERO_t$, respectivamente. Las series de tiempo de la lluvias medias, expresadas en litros por metro cuadrado y día, (l/m^2d), de los pluviómetros de Fervenza, Puente Olveira y Santa Eugenia se denotan por MFR_t, $MSER_t$, $MPOR_t$, respectivamente. La figura 2.2 muestra las series de precipitaciones medias en Fervenza y Santa Eugenia.

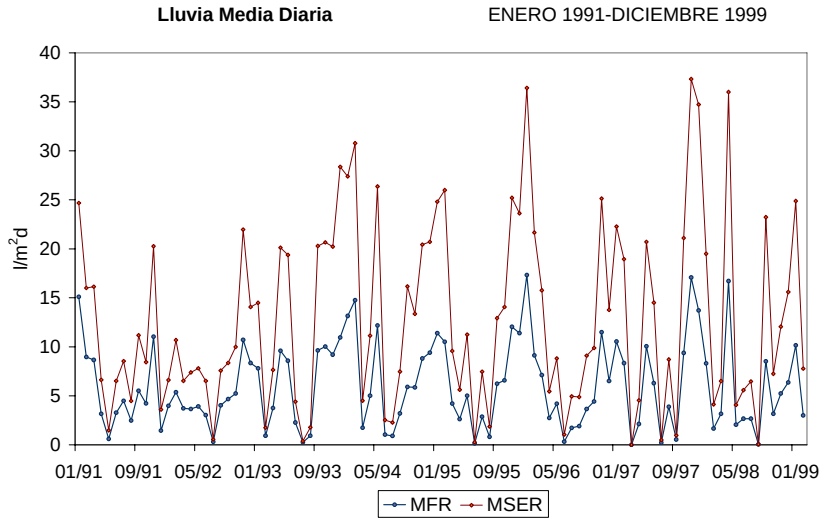

Figura 2.2 Series de Lluvias medias en Fervenza y Santa Eugenia.

2.2 Modelización mensual de las aportaciones. Modelos Box-Jenkins

2.2.1 Breve Introducción a las Series de Tiempo

En ocasiones resulta necesario el estudio de la evolución de una series de datos obtenidos a intervalos de tiempo regulares, series de tiempo (Shumway y Stoffer, 2000; Brockwell y Davis, 2002; Peña, 2005). El estudio de estas series puede realizarse desde dos puntos de vista diferentes: el análisis univariante, en el que sólo se considera la evolución previa de la serie o inercia, o los modelos de regresión dinámica en los que la evolución de la serie de tiempo se ve influida por otras variables.

El modelo matemático de una serie de tiempo se denomina proceso estocástico. Si se considera el valor observado de la serie en el instante *t* como una observación de una variable aleatoria, una serie de tiempo con *n* valores, será una observación de *n* variables aleatorias ordenadas en orden cronológico. Resulta habitual establecer hipótesis relativas a la distribución conjunta de las variables con el fin de establecer un modelo que se comporte como el proceso de generación de la serie de tiempo. Los modelos más populares son los Box-Jenkins(Box y Jenkis, 1976; Box *et al*., 1994; Mack, 1996). Dentro de esta clase de modelos se hará especial hincapié en los modelos Autorregresivos (AR), y los de Media Móvil (MA).

La variable a predecir asociada a un instante *t*, Z_t, en un modelo autorregresivo, AR, de orden *p*, tiene dependencia lineal sobre los *p* términos anteriores de la serie. Pude considerarse el Operador Retardo, *B*, que funciona según (2.1). De este modo un modelo AR(*p*) puede ser expresado en la forma (2.2).

$$\begin{aligned} &BZ_t = Z_{t-1}, \text{ for all } t \in \mathbb{Z} \\ &Bc = c, \text{ for all } c \in \mathbb{R}, \text{constant} \end{aligned} \quad . \tag{2.1}$$

$$Z_t = a_1 Z_{t-1} + \cdots + a_p Z_{t-p} + \varepsilon_t = a_1 B Z_t + \cdots + a_p B^p Z_t + \varepsilon_t = \left(\sum_{i=1}^{p} a_i B^i \right) Z_t + \varepsilon_t \tag{2.2}$$

siendo ε_t la serie de *ruido* que tiene una distribución normal de media cero y varianza $\sigma^2(t)$ constante (2.3). De este modo el modelo AR(p) puede expresarse como (2.4).

$$\varepsilon_t \in N\left(0, \sigma(t)\right) \tag{2.3}$$

$$\left(1 - \psi_p(B)\right) Z_t = \varepsilon_t \tag{2.4}$$

siendo ψ_p un polinomio de orden *p en el operador B* de la forma $\psi_p = \sum_{i=1}^{p} a_i B^i$.

Los modelo de media móvil, MA, de orden *q*, suponen que el comportamiento de Z_t presenta dependencia lineal respecto de los últimos *q* valores de la serie de ruido. Puede expresarse como (2.5) o equivalentemente (2.6).

$$Z_t = m_1 \varepsilon_{t-1} + \cdots + m_q \varepsilon_{t-q} + \varepsilon_t = m_1 B \varepsilon_t + \cdots + m_q B^q \varepsilon_t + \varepsilon_t = \left(\sum_{j=1}^{q} m_j B^j \right) \varepsilon_t + \varepsilon_t \tag{2.5}$$

$$Z_t = \left(1 + \phi_q\left(B\right)\right)\varepsilon_t \tag{2.6}$$

siendo ϕ_q un polinomio de orden *q* en *B*.

Es posible considerar los modelos ARMA, que se construyen a partir de la combinación ambos modelos, AR y MA, y presentan la forma (2.7)

$$Z_t = \sum_{i=1}^{p} a_i \cdot Z_{t-i} + \sum_{j=1}^{q} m_j \cdot \varepsilon_{t-j} + \varepsilon_t \tag{2.7}$$

Un modelo Box-Jenkins más general, el modelo Autorregresivo Integrado de Media Móvil (ARIMA) (Wei, 1990; Makridakis *et al.* 1998), surgen por la necesidad de explicar series de tiempo con comportamiento periódico y de tendencia no constante. Se incorpora la idea de estudiar la serie de las diferencias, en lugar de la serie original. Las diferencias locales se definen por (2.8), mientras que las diferencias periódicas se definen por (2.9), siendo ∇ el operador retardo (Backward Difference Operator) y p el retardo temporal asociado a la periodicidad.

$$\nabla Z_t = \left(1 - B\right) Z_t = Z_t - Z_{t-1} \tag{2.8}$$

$$\nabla_p Z_t = \left(1 - B^p\right) Z_t = Z_t - Z_{t-p} \tag{2.9}$$

El esquema de un modelo ARIMA estacional, $ARIMA\left(a,b,c\right)\times\left(d,e,f\right)p$ es

$$\nabla_p^e \nabla^b Z_t = \sum_{i=1}^{a} a_i \cdot Z_{t-i} + \sum_{i=1}^{c} c_i \cdot \varepsilon_{t-i} + \sum_{i=1}^{d} d_i \cdot Z_{t-i\cdot p} + \sum_{i=1}^{f} f_i \cdot \varepsilon_{t-i\cdot p} + \varepsilon_t, \text{con } \varepsilon_t \in N\left(0, \sigma\left(t\right)\right) \tag{2.10}$$

El significado de los parámetros es el siguiente

- a: el orden del término AR local o regular
- b: el número de diferencias locales
- c: el orden del término MA local o regular
- d: el orden del término AR periódico
- e: el número de diferencias periódicas
- f: el orden del término MA periódico
- p: el número de retardos asociados a la periodicidad de la serie

Cuando el comportamiento de una serie de tiempo no puede ser satisfactoriamente explicado, por el análisis univariante de series de tiempo, se puede considerar la incorporación de variables regresoras, llamadas series exógenas. La regresión dinámica (ARIMAX) combina los modelos Box-Jenkins con la regresión lineal, obteniendo un modelo más general para el estudio

de las series de tiempo (Shumway y Stoffer, 2000). Si se desea explicar $\{Z_t\}$, usando la serie exógena $\{X_t\}$ se obtiene un modelo que responde a (2.11).

$$\nabla_p^e \nabla^b Z_t = \sum_{k=0}^{r} r_i \cdot X_{t-k} + \sum_{i=1}^{a} a_i \cdot Z_{t-i} + \sum_{i=1}^{c} c_i \cdot \varepsilon_{t-i} + \sum_{i=1}^{d} d_i \cdot Z_{t-i \cdot p} + \sum_{i=1}^{f} f_i \cdot \varepsilon_{t-i \cdot p} + \varepsilon_t \qquad (2.11)$$

Para todos estos modelos de análisis de serie de tiempo, bajo la hipótesis de normalidad, pueden construirse intervalos de predicción. Ésta es una de las ventajas del acercamiento estadístico frente al acercamiento a través de modelos matemáticos, la posibilidad de construir intervalos de predicción y confianza, basados en la presunción de normalidad y en la estructura del modelo de serie de tiempo. Estos intervalos de predicción proporcionan límites de seguridad al modelo y permiten asociar una cantidad significativa de datos fuera de los intervalos de predicción con cambios de la estructura física del sistema hidrológico en estudio, cambios que tienen consecuencias serias en el funcionamiento del modelo de serie de tiempo, y hará necesario una reevaluación de la estructura modelo y de los parámetros del mismo.

2.2.2 Los modelos seleccionados

Se ha realizado un estudio completo de la escorrentía o caudal medio mensual medido en las presas de Fervenza y Santa Eugenia. La selección de modelos y la estimación de los parámetros fue hecha a partir de los datos registrados entre enero de 1991 y agosto de 1999. El funcionamiento de los modelos fue evaluado usando los datos mensuales desde septiembre de 1999 a septiembre de 2000, aproximadamente el último el 10% de los datos disponibles. Se diseñó una aplicación en MS.Excel cada modelo, con predicción multirretardo de hasta 15 datos, para hacerlo disponible para el usuario en futuras predicciones.

El número máximo de retrasos, quince, fue seleccionado para obtener las predicciones del nuevo año hidrológico con tres meses de adelanto. Estos tres meses proporcionan el tiempo suficiente para preparar los proyectos de explotación para el próximo año. La aplicación tenía además disponible la construcción de intervalos de predicción, con nivel de confianza seleccionado por el usuario. Los modelos desarrollados se realizaron sobre las variables originales o sobre transformaciones Box-Cox (Box y Cox, 1964) de las mismas, y sobre variables estandarizadas. Para cada caso se construyeron dos modelos, considerando o no la información proporcionada por la serie de precipitaciones. De este modo se construyeron 4 modelos para las aportaciones medidas en cada presa, con el fin de comparar modelos con diferentes niveles de complejidad, desde los más sencillos hasta otros más elaborados, que involucran variables exógenas.

2.2.2.1 El Primer Modelo

Una primera aproximación al problema ha sido emplear un modelo *autoexplicativo*, esto es, que base su predicción únicamente en las aportaciones del pasado, sin considerar la información adicional proporcionada por la lluvia recogida en los diferentes medidores pluviométricos.

Durante el estudio de las aportaciones mensuales tanto en Fervenza como en Santa Eugenia se ha observado que los errores no eran homocedásticos, por lo que se realizó una

transformación logarítmica de los datos, a fin de estabilizar la varianza (Box y Cox, 1964). Se modelaron entonces las dos series transformadas, (2.12) y (2.13).

$$\tilde{X}_t = \log(MMFRO_t) \tag{2.12}$$

$$\tilde{Z}_t = \log(MMSERO_t) \tag{2.13}$$

El modelo Box-Jenkins seleccionado para las aportaciones de Fervenza es un $ARIMA(0,0,1)\times(1,1,1)12$,(2.14), siendo *k* una constante que permite considerar la evolución de los valores medios de los datos. Se puede observar que el modelo recoge la estacionalidad anual del clima, y en consecuencia de las aportaciones registradas.

$$\tilde{X}_t = \tilde{X}_{t-12} + c_1 \cdot \varepsilon_{t-1} + f_1 \cdot \varepsilon_{t-12} + \varepsilon_t + k \tag{2.14}$$

El modelo Box-Jenkins seleccionado para las aportaciones medias mensuales en Santa Eugenia, sigue un modelo $ARIMA(0,0,1)\times(1,1,1)12$, (2.15). De nuevo el modelo seleccionado refleja la periodicidad climática inherente al problema.

$$\tilde{Z}_t = (1+d_1)\cdot\tilde{Z}_{t-12} + c_1 \cdot \varepsilon_{t-1} + f_1 \cdot \varepsilon_{t-12} + \varepsilon_t + k \tag{2.15}$$

2.2.2.2 El Segundo Modelo

Frente al modelo autoexplicativo univariante, surge la idea de utilizar la lluvia registrada en los pluviómetros de Fervenza, Puente Olveira y Santa Eugenia como fuente de información, por la más que evidente relación que existe entre las precipitaciones y las aportaciones. Se han modelizado, entonces, las aportaciones en las dos presas, empleando un modelo que combine la modelización Box-Jenkins con la regresión lineal sobre la lluvia registrada, esto es, usando la serie de lluvia como variable exógena. Se sigue deseando proporcionar predicciones multirretardo, y por lo tanto es necesario modelizar el comportamiento de los registros de lluvia en los distintos pluviómetros a fin de incorporar las predicciones de lluvia al modelo que estima las aportaciones.

La evolución del comportamiento de la lluvia ha sido en los tres pluviómetros de los que disponemos ha sido estudiada empleando modelización Box-Jenkins. Los modelos obtenidos para la lluvia son los siguientes. La serie temporal de la lluvia (*l/m²d*) recogida en el pluviómetro situado en la presa de Fervenza, MRF_t, sigue un modelo $ARIMA(1,0,0)\times(1,0,0)12$ con constante; la lluvia de Santa Eugenia, $MRSE_t$, presenta un comportamiento que se puede modelizar a través de un $ARIMA(1,0,0)\times(1,0,0)12$ con constante, (2.17), igual que el anterior. La lluvia registrada en Puente Olveira presenta un comportamiento mucho más variable, por lo que ha sido necesario realizar una transformación logarítmica de la serie, asociada a una traslación para evitar los problemas inherentes al cero frente a funciones logarítmicas. De este modo se ha pasado a estudiar la serie transformada $Y_t=log(1+MRF_t)$, cuyo comportamiento responde a un $ARIMA(0,0,0)\times(1,0,1)12$ con constante, (2.18).

$$MFR_t = a_1 \cdot MFR_{t-1} + d_1 \cdot MFR_{t-12} + \varepsilon_t + k \tag{2.16}$$

$$MSER_t = a_1 \cdot MSER_{t-1} + d_1 \cdot MSER_{t-12} + \varepsilon_t + k \tag{2.17}$$

$$Y_t = a_1 \cdot Y_{t-12} + f_1 \cdot \varepsilon_{t-12} + \varepsilon_t + k \quad (2.18)$$

Una vez modelizadas las series de lluvias se ha estudiado la dependencia entre las series de lluvia y las de aportaciones. Los modelos resultantes fueron los siguientes. Las aportaciones de Fervenza pueden explicarse a través de un modelo $ARIMA(0,0,0)\times(1,0,0)12$,(2.19),con constante, y con tres variables regresoras que han resultado diferentes retardos de la lluvia en Fervenza; la lluvia en el mes que se desea predecir, MRF_t, en el mes previo, MRF_{t-1}, y el año anterior, MRF_{t-12}.

$$X_t = d_1 \cdot X_{t-12} + r_0 \cdot MFR_t + r_1 \cdot MFR_{t-1} + r_{12} \cdot MFR_{t-12} + \varepsilon_t + k \quad (2.19)$$

Este modelo refleja el comportamiento periódico tanto de la escorrentía como de las precipitaciones. El primer término de retardo refleja las características físicas de la subcuenca de Fervenza. Es una subcuenca lenta, por lo que al agua procedente de la lluvia tarda cierto tiempo en transformarse en caudal en el río.

Por su parte las aportaciones de Santa Eugenia siguen un modelo $ARIMA(1,0,0)$ con constante, (2.20), y con dos variables regresoras, la lluvia en el mes de predicción en Puente Olveira, $MRPO_t$, y en Fervenza, MRF_t.

$$Z_t = a_1 \cdot Z_{t-1} + r_0 \cdot MFR_t + s_0 \cdot MSER_t + \varepsilon_t + k \quad (2.20)$$

Se puede observar la desaparición de la componente estacional. La información local, esto es, reciente, resulta suficiente para explicar la evolución de la serie. En este caso no se encuentran involucrados retardos en los términos de lluvia, debido a que la subcuenca de Santa Eugenia presenta una respuesta más rápida que la subcuenca de Fervenza.

2.2.2.3 El Tercer Modelo

Habiéndose observado en el primer enfoque del problema que ni la serie de aportaciones de Fervenza ni la de Santa Eugenia son homocedásticas se ha considerado la idea de estandarizar las series de las aportaciones, y estudiar el comportamiento de estas nuevas series a las que se denota por $SMAF_t$, y $SMASE_t$, respectivamente. Se ha enfocado el estudio de estas series de modo univariante (esto es, sin información exógena) de modo que los modelos para ambas series, las series medias mensuales de aportaciones de Fervenza,(2.21), y Santa Eugenia,(2.22), estandarizadas, responden a un $ARIMA(1,0,0)$ con constante. Las estructuras de las ecuaciones asociadas son idénticas y simples,(2.23).

$$\tilde{\tilde{X}}_t = SMMFRO_t \quad (2.21)$$

$$\tilde{\tilde{Z}}_t = SMMSERO_t \quad (2.22)$$

$$\tilde{\tilde{Y}}_t = a_1 \cdot \tilde{\tilde{Y}}_{t-1} + \varepsilon_t + k \quad (2.23)$$

Tanto para $\tilde{\tilde{Y}}_t = \tilde{\tilde{X}}_t$ como para $\tilde{\tilde{Y}}_t = \tilde{\tilde{Z}}_t$. Las variables estandarizadas han perdido su componente periódica.

2.2.2.4 El Cuarto Modelo

Tal y como se hizo con el segundo modelo, se ha buscado un modelo de regresión dinámica que se apoye en la información pluviométrica disponible. Se ha llevado a cabo el estudio de las series estandarizadas de las aportaciones en Fervenza y en Santa Eugenia, considerando la influencia que pueden tener en ellas las series de lluvias MRF_t, $MRPO_t$, y $MRDE_t$. De este modo se obtienen los siguientes modelos.

Las aportaciones medias mensuales estandarizadas medidas en la presa de Fervenza siguen un modelo $ARIMA(1,0,0)$ con constante y con dos variables regresoras, que serán las precipitaciones recogidas en el mes que se desea predecir en Santa Eugenia, $MRDE_t$, y en Fervenza, MRF_t. Luego el modelo responde a (2.24).

$$\tilde{\tilde{X}}_t = a_1 \cdot \tilde{\tilde{X}}_{t-1} + r_0 \cdot MFR_t + s \cdot MSER_t + \varepsilon_t + k \qquad (2.24)$$

Si se observa la estructura de la cuenca, Figura 2.1, las precipitaciones recogidas en Santa Eugenia no pueden verse físicamente reflejadas en Fervenza, pues Fervenza está aguas arriba, luego la lluvia de Santa Eugenia parece reflejar las precipitaciones de una localización indefinida aguas arriba de la presa de Fervenza.

Las aportaciones estandarizadas medias mensuales en Santa Eugenia pueden ser modelizadas siguiendo un $ARIMA(1,0,0)\times(1,0,0)12$ con constante y con dos variables regresoras que han resultado ser las precipitaciones recogidas en Santa Eugenia y Puente Olveira en el mes que se desea predecir. El modelo es pues, (2.25). El modelo sugiere que la subcuenca inferior del río no recibe aportaciones significativas procedentes de la lluvia en la subcuenca superior.

$$\tilde{\tilde{Z}}_t = a_1 \cdot \tilde{\tilde{Z}}_{t-1} + d_1 \cdot \tilde{\tilde{Z}}_{t-12} + r_0 \cdot MPOR_t + s \cdot MSER_t + \varepsilon_t + k \qquad (2.25)$$

En este modelo ha reaparecido el comportamiento periódico en la estructura de los datos.

2.3 Modelización diaria de las aportaciones. Redes neuronales frente a Modelos Box-Jenkins

Se consideran en este estudio dos horizontes de predicción, tal y como se comentaba en la introducción. El horizonte a medio plazo o mensual ha sido satisfecho ampliamente con el uso de modelos clásicos de predicción de series temporales, pero al estudiar el problema a corto plazo se ha decidido enfocar el problema desde dos perspectivas diferentes, con el fin de comparar la acción de las redes neuronales en la predicción de variables continuas en series temporales, con la de los modelos Box-Jenkins. Los modelos Box-Jenkins han sido comentados en la sección anterior, por lo que en esta sección se dedicará un apartado a ampliar algunos de los conceptos introducidos en el capítulo inicial de la presente tesis.

2.3.1 Detalles sobre Redes Neuronales Artificiales.

En el primer capítulo se realizó una introducción a las redes neuronales, sus objetivos, metodologías, estructuras y propiedades. En esta sección se reforzarán algunas ideas, asociadas a la predicción de variables continuas en general, y aplicables al problema de predicción que se trata en particular.

La estructura de red neuronal que se decidió emplear en este caso es la de una red con una sola capa oculta, por todas las razones expuestas en la sección 1.7. En todos los problemas modelizados a lo largo de esta tesis se emplean redes neuronales off-line, debido a la inestabilidad inherente a las redes on-lines. En cualquier caso, según las características del problema, y el grado y la velocidad de cambio de las propiedades del sistema estudiado será necesario realizar reentrenamientos de la red de modo periódico. Tanto la frecuencia del reentrenamiento como la selección del conjunto de aprendizaje empleado son asuntos que requieren ser tratados cuidadosamente. Una frecuencia excesiva en el reentrenamiento, sobre todo en situaciones de estabilidad del sistema, pueden dar lugar a conjuntos de entrenamiento poco representativos del funcionamiento global del sistema, (centrados sólo en un estado estable concreto) que generen redes "ciegas" al resto de posibilidades del sistema, aunque muy exactas en la predicción del comportamiento presente del sistema en estudio. Este es un mal habitual que puede ser tratado con la construcción de matrices históricas de datos destinadas a almacenar datos de las diversas situaciones en las que se ha encontrado el sistema, renovando el conjunto de datos pero sin perder la información de los episodios anómalos y particulares que se han producido. Este mismo problema se puede presentar, como se comentó en el capítulo 1 si el entrenamiento, que deberá online y no batch (Bishop, 1995), en sí mismo no sigue una secuenciación aleatoria, debido a que la red se ajustará favorablemente a los datos más recientes, los últimos en presentarse a la red, produciéndose un sobreentrenamiento en relación a estos datos; el sobreentrenamiento es un problema (Mitchell, 1997) que puede evitarse seleccionando adecuadamente el conjunto de entrenamiento y el método de entrenamiento, además de la relación entre el número de parámetros de la red y el número de datos disponibles. El algoritmo de entrenamiento empleado será backpropagation, en su versión online, la regla delta, (Widrow y Hoff, 1960, Cheng y Titterington, 1994), para una función de error. En este caso la variable a estudiar y predecir es una variable continua, lo que hace que el error cuadrático medio se a una elección adecuada para la función de error de la red.

Es importante no confundir la función de error con la función de bondad de ajuste que se empleará para estudiar y comparar el ajuste del modelo. En la literatura hidrológica se han considerado diversas funciones para evaluar la bondad de ajuste de los modelos,(Garrick *et al.*, 1978, Legates y McGabe, 1999; Kneale *et al.*, 2001), desde el coeficiente de eficiencia, al índice de concordancia, o el coeficiente de determinación, así como variantes de éstos. En la sección 2.3 se definirán algunas medidas de bondad de ajuste, alguna muy popular, como el error medio absoluto, MAE. Todos ellos han sido ampliamente empleados, y son funciones de error estadísticamente muy conocidas.

Este estudio no sólo se refiere a variables continuas, sino también a variables que constituyen series de tiempo. El enfoque del estudio de series de tiempo a través de redes neuronales ha sido abordado de diversas formas, según los objetivos planteados (Azoff, 1994; Whitehead y Choate, 1996). Si se desea predecir un único retardo las redes neuronales estáticas feedforward son adecuadas para el proceso, tal y como se puede apreciar a través de este estudio. Pero si se plantea un horizonte multirretardo existen distintas posibilidades para llevar a buen término dicho propósito. Una propuesta acertada consiste en emplear una capa de salida para la red neuronal multinodo, de modo que cada nodo prediga, y por tanto se

compare, con el dato del horizonte correspondiente que se desea predecir. Será pues una predicción múltiple y en cierta medida independiente entre los distintos horizontes de predicción. Existen asimismo otras posibilidades. Se puede definir un tipo de red, llamado recurrente, en el que la capa de salida establece un nexo con la entrada, de modo que las salidas de la red sean incorporadas como entrada para la predicción del siguiente retardo. De esta manera la predicción a un horizonte dado involucra de modo recurrente a las predicciones de los horizontes anteriores. El entrenamiento de estas redes es ligeramente distinto, pues se van las observaciones en el conjunto de datos generadas para la estimación de posteriores horizontes variarán a medida que se entrene la red neuronal.

En este caso se emplearán redes clásicas, llamadas estáticas, para la predicción. En este caso se han empleado dos estructuras diferentes para las redes neuronales, con el fin de poner de manifiesto la necesidad de una adecuada selección de la arquitectura de la red.

2.3.2 Los Datos Diarios

Se dispone de los datos diarios de aportaciones comprendidos entre el 16 de enero de 1991 y el 29 de Junio del 2000. Se han estimado los modelos Box-Jenkins y la red neuronal empleando los datos de lluvia y escorrentía de este período. Para evaluar el funcionamiento de ambos métodos se han analizado las predicciones obtenidas en el conjunto de validación reservado, que abarca desde el 16 de Noviembre del 2000 al 31 de Enero del 2001. La selección del conjunto de validación se hizo teniendo en cuenta que el período que abarca desde mediados de noviembre a enero corresponde al período mas problemático, en términos de lluvia, del año en la cuenca del Xallas. En particular este conjunto de validación fue una de las temporadas más lluviosas de la historia reciente de Galicia, de tal modo que no se tiene constancia de lluvias similares en más de medio siglo, de modo que obtener predicción adecuadas era una tarea difícil. Por eso mismo los resultados en la predicción fueron muy importantes.

Las variables de entrada fueron medidas correspondientes a la presa de Fervenza. En primer lugar se consideró la lluvia acumulada de los 15 días previos, (2.26). Esta variable intenta condensar la información relativa a la humedad presente en la capa freática. Se ha seleccionado así mismo la lluvia en Fervenza del día anterior, (2.27), la predicción de lluvias en el día de predicción, (2.28), y las aportaciones medias del día anterior, (2.29). La variable de salida o predicción es la aportación media en Fervenza en el instante *t*, (2.30).

$$Z_{1t} = DA15FR_{t\text{-}1} = \sum_{i=1}^{15} DFR_{t-i} \tag{2.26}$$

$$Z_{2t} = DFR_{t\text{-}1} \tag{2.27}$$

$$Z_{3t} = DFR_t \tag{2.28}$$

$$Z_{4t} = MDFRO_{t-1} \tag{2.29}$$

$$Z_{5t} = MDFRO_t \tag{2.30}$$

Tanto las variables de salida como las de entrada han sido transformadas según (2.31).Se denotaron a las variables transformadas de entrada X_i, *con i=1,2,3,4*. La variable objetivo se denota por X_5.

$$X = \log(Z + 1) \tag{2.31}$$

Esta transformación fue considerada necesaria por la variabilidad de los datos. La lluvia muestra a menudo un comportamiento caótico, y la transformación propuesta permite suavizar los datos. La traslación como ya ocurría en alguno de los datos mensuales es debida a los problemas que presenta el logaritmo en el cero. El análisis de datos muestra la alta dependencia de la escorrentía de un día con la lluvia y la escorrentía del día anterior. La idea de añadir como término la lluvia acumulada surge de la búsqueda de una variable que refleje de modo realista la situación de humedad del sistema hidrológico. No se han empleado datos de lluvia correspondientes a otros pluviómetros porque la gran dependencia entre las distintas variables de lluvia podría constituir una fuente de problemas durante el problema de entrenamiento. La correlación entre la lluvia de Fervenza y Santa Eugenia alcanza el valor de *0.88*, entre Fervenza y Puente Olveira llega a *0.96*, y entre Santa Eugenia y Puente Olveira es de *0.91*.

2.3.3 El Modelo de Red Neuronal Artificial Propuesto

Se ha empleado un perceptrón multicapa con una capa oculta, otro con una capa oculta funcional. Ambas arquitecturas proporcionan aproximadores universales (Chen y Manry, 1993). El perceptrón con una capa funcional es una red basada en las propiedades de la familia funcional asociada a la capa oculta. El origen de esta idea se basa en la existencia de familias de funciones que generan espacios de funciones suaves (Harmuth, 1972), tal y como se comentaba en el capítulo1. La relación entre las entradas y las salidas se supone que pertenece a un espacio de funciones suaves, en el que la familia que genera la capa oculta es densa. La exactitud de la aproximación depende directamente del número de funciones de la familia seleccionadas, que a su vez viene establecido por el número de nodos de la capa oculta. Cada dato de entrada es transformado de modo individual por todo el subconjunto de funciones, de modo que si se considera un subconjunto de S funciones y se tienen N_I variables de entrada, el número de nodos de la capa oculta será (2.32).

$$N_H = N_I \cdot S \tag{2.32}$$

En este caso se ha empleado una base de funciones polinómicas, de modo que se tiene una capa funcional polinómica. El grado del polinomio coincide con S. La función de transferencia en la capa de salida es la identidad. Cada subconjunto de funciones es la función de transferencia de N_I nodos de la capa oculta. Los pesos entre la capa de entrada y la oculta valen 1 y no serán cambiado durante el entrenamiento. El valor de S aumentará hasta que el aumento de S no genere una mejora en el funcionamiento de la red.

El modelo seleccionado fue un perceptrón con una capa polinómica de grado S=2. El algoritmo de backpropagation fue el empleado par el entrenamiento, al disponer de una función de error y funciones de activación diferenciables, (Rumelhart, 1986). La salida de la red responde a (2.33), donde los X_i responden a (2.31). Esto hace que para la estimación de las aportaciones

originales en Fervenza sea necesario deshacer el cambio, esto es, invertir la transformación logarítmica,(2.34). La predicción de la escorrentía media diaria, será entonces (2.35)

$$o_{NN} = \sum_{j=1}^{S}\sum_{i=1}^{4} \omega^{ho}_{(i-1)\cdot S+j\,1} \cdot X_i^j + \omega^{ho}_{01} \tag{2.33}$$

$$Z = e^X - 1 \tag{2.34}$$

$$\hat{Z}_5 = \tilde{o}_{NN} = e^{o_{NN}} - 1 \tag{2.35}$$

2.3.4 El Modelo Box-Jenkins Propuesto

El análisis de las funciones de autocorrelación y autocorrelación parcial hicieron que se seleccionase un modelo AR(1), que fue completado con las variables Z_1, Z_2, Z_3, definidas en la subsección anterior. El modelo resultante puede expresarse como (2.36). Considerando la definición de las variables, se tienen (2.37), (2.38) y (2.39).

$$Z_{5t} = a_1 Z_{4t} + r_1 \cdot Z_{1t} + r_2 \cdot Z_{2t} + r_3 \cdot Z_{3t} + \varepsilon_t \tag{2.36}$$

$$Z_{5(t-1)} = Z_{4t} \tag{2.37}$$

$$Z_{3(t-1)} = Z_{2t} \tag{2.38}$$

$$Z_{1t} = \sum_{i=1}^{15} Z_{3(t-i)} \tag{2.39}$$

De este modo el modelo resultante responde a (2.40) o equivalentemente a (2.41). La predicción de Z_5 será (2.42).

$$Z_{5t} = a_1 Z_{5(t-1)} + r_1 \cdot \sum_{i=1}^{15} Z_{3(t-i)} + r_2 \cdot Z_{3(t-1)} + r_3 \cdot Z_{3t} + \varepsilon_t \tag{2.40}$$

$$Z_{5t} = a_1 Z_{5(t-1)} + r_1 \cdot \sum_{i=-21}^{15} Z_{3(t-i)} + (r_1 + r_2) \cdot Z_{3(t-1)} + r_3 \cdot Z_{3t} + \varepsilon_t \tag{2.41}$$

$$\tilde{o}_{BJ} = a_1 Z_{5(t-1)} + r_1 \cdot \sum_{i=-21}^{15} Z_{3(t-i)} + (r_1 + r_2) \cdot Z_{3(t-1)} + r_3 \cdot Z_{3t} \tag{2.42}$$

2.4. Resultados y Discusión

2.4.1 Resultados Mensuales

Se emplearon cuatro modelos diferentes para la predicción de aportaciones. Se ha estudiado el comportamiento de los diferentes modelo, y en la práctica se ha transferido a los usuarios el modelo que mejor se comportaba en cada caso, que ha resultado depender del mes que se desea predecir, de modo que se ha construido un modelo combinado.

Para comparar los modelos se han considerado dos medidas de error. Por una parte se tiene el error medio absoluto, *MAE,*(2.43), y máximo del error absoluto, *MaxAE,* (2.44), cometido por el modelo. En estas expresiones $\{Y_t\}$ es la serie a estimar y $\{\hat{Y}_t\}$ la serie de las estimaciones.

$$MAE = \frac{1}{N}\sum_{t=1}^{N} \left| Y_t - \hat{Y}_t \right| \tag{2.43}$$

$$MaxAE = \max\left|Y_t - \hat{Y}_t\right| \tag{2.44}$$

Las tablas 2.1 a 2.4 muestran el error de cada modelo en cada mes. La tabla 2.5 resumen al información de las tablas anteriores.

MaxAE	*ENE*	*FEB*	*MAR*	*ABR*	*MAY*	*JUN*	*JUL*	*AGO*	*SEP*	*OCT*	*NOV*	*DEC*
MODELO 1	25.70	17.31	24.31	31.22	14.05	5.28	10.78	1.29	3.61	20.75	17.62	**13.87**
MODELO 2	**16.66**	**12.57**	19.97	**8.66**	11.58	8.41	13.78	5.62	10.88	15.33	**10.68**	14.89
MODELO 3	23.68	13.30	20.96	25.58	12.83	**4.85**	**8.39**	**1.14**	3.56	10.43	16.13	22.74
MODELO 4	19.80	17.35	**19.46**	16.10	**9.70**	5.16	10.35	1.37	**2.91**	**7.22**	18.49	28.29

Tabla 2.1 Máximo error de los diferentes modelos en cada mes en Fervenza

MAE	*ENE*	*FEB*	*MAR*	*ABR*	*MAY*	*JUN*	*JUL*	*AGO*	*SEP*	*OCT*	*NOV*	*DEC*
MODELO 1	12.49	7.28	7.73	7.95	6.41	2.70	1.85	0.73	1.53	5.96	8.25	8.32
MODELO 2	**8.89**	**5.77**	8.04	**5.03**	4.77	3.43	35.34	3.30	3.49	6.41	**5.57**	8.94
MODELO 3	12.88	6.69	7.65	10.24	5.13	2.26	**1.79**	0.51	1.47	5.57	6.64	**8.10**
MODELO 4	10.42	6.19	**6.81**	7.94	**4.11**	**1.59**	2.26	**0.50**	**1.21**	**4.60**	7.85	13.60

Tabla 2.2 Error absoluto medio de los diferentes modelos en cada mes en Fervenza

MaxAE	*ENE*	*FEB*	*MAR*	*ABR*	*MAY*	*JUN*	*JUL*	*AGO*	*SEP*	*OCT*	*NOV*	*DEC*
MODELO 1	21.94	15.77	17.70	25.03	15.29	5.56	0.99	**1.50**	5.97	13.27	10.95	**16.98**
MODELO 2	17.65	19.12	**12.86**	19.51	14.92	**4.89**	2.51	4.53	6.24	14.03	10.98	21.30
MODELO 3	22.81	17.97	15.53	**11.00**	12.98	9.81	15.14	6.04	10.10	13.57	**9.64**	18.85
MODELO 4	**14.09**	**11.76**	13.17	11.01	**12.39**	4.93	**0.93**	1.90	**3.74**	**9.83**	14.12	34.43

Tabla 2.3 Máximo error de los diferentes modelos en cada mes en Santa Eugenia

MAE	*ENE*	*FEB*	*MAR*	*ABR*	*MAY*	*JUN*	*JUL*	*AGO*	*SEP*	*OCT*	*NOV*	*DEC*
MODELO 1	9.76	7.31	5.88	7.58	5.06	1.80	**0.42**	0.61	2.42	4.88	7.95	9.68
MODELO 2	8.22	6.35	7.02	5.36	7.12	3.25	3.41	3.78	2.98	**4.76**	6.52	12.69
MODELO 3	9.11	7.33	7.58	4.71	6.13	4.10	3.36	2.74	3.40	5.41	**4.72**	**8.28**
MODELO 4	**7.45**	**5.11**	**4.00**	**4.56**	**3.71**	**1.50**	0.53	**0.56**	**1.77**	5.18	7.44	15.50

Tabla 2.4 Error absoluto medio de los diferentes modelos en cada mes en Santa Eugenia

MEJOR MODELO		*ENE*	*FEB*	*MAR*	*ABR*	*MAY*	*JUN*	*JUL*	*AGO*	*SEP*	*OCT*	*NOV*	*DEC*
F.	MaxAE	M. 2	M. 2	M. 4	M. 2	M. 4	M. 3	M. 3	M. 3	M. 4	M. 4	M. 2	M. 1
	MAE	M. 2	M. 2	M. 4	M. 2	M. 4	M. 4	M. 3	M. 4	M. 4	M. 4	M. 2	M. 3
S.E	*MaxAE*	M. 4	M. 4	M. 2	M. 3	M. 4	M. 2	M. 4	M. 1	M. 4	M. 4	M. 3	M. 1
	MAE	M. 4	M. 4	M. 4	M. 4	M. 4	M. 4	M. 1	M. 4	M. 4	M. 2	M. 3	M. 3

Tabla 2.5 Tabla resumen. Mejor modelo para cada presa y cada criterio

Las figuras 2.3 a 2.6 muestran las comparaciones de las series observadas y las predicciones realizadas con los modelos 1 a 4, desde enero de 1992 a septiembre de 2000. A causa de la estructura periódica de algunos modelos, no se obtuvieron predicciones de los primeros datos, por lo que se omitieron los datos del año 1991 de las gráficas. Los primeros modelos de Fervenza y Santa Eugenia son los más sencillo, y los cuartos los de regresión dinámica, más complejos. El período de validación es el último año, y el período anterior es el empleado para estimar los parámetros.

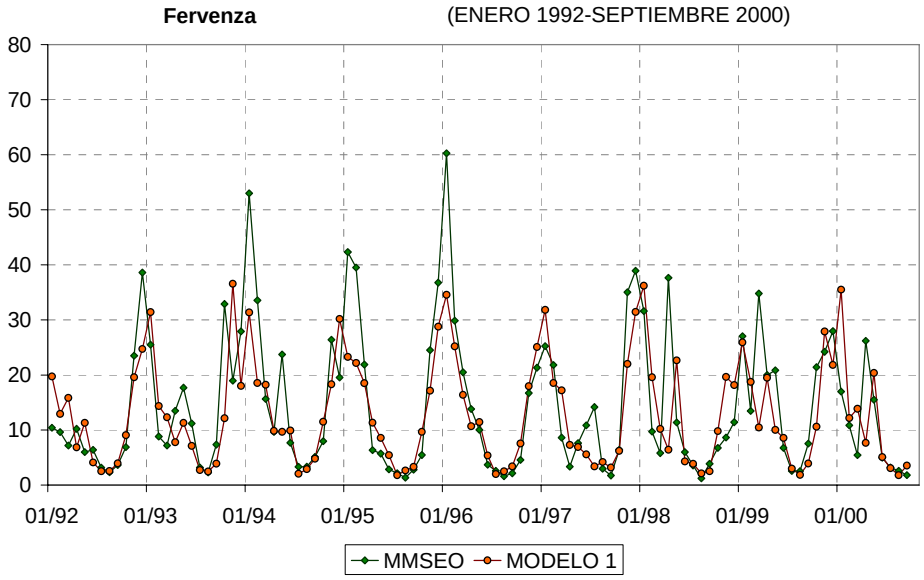

Figura 2.3. Predicción del Modelo 1 frente a la serie real. Fervenza

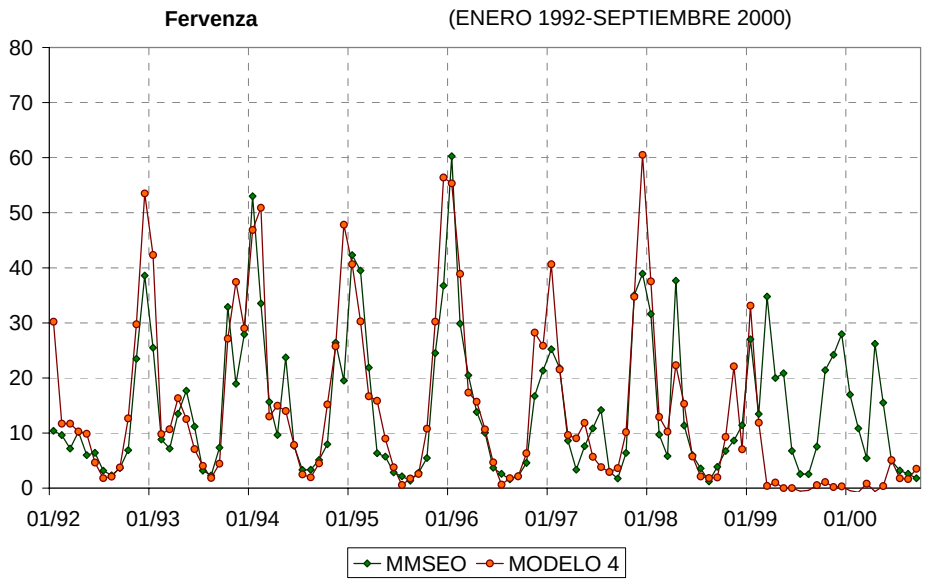

Figura 2.4. Predicción del Modelo 4 frente a la serie real. Fervenza

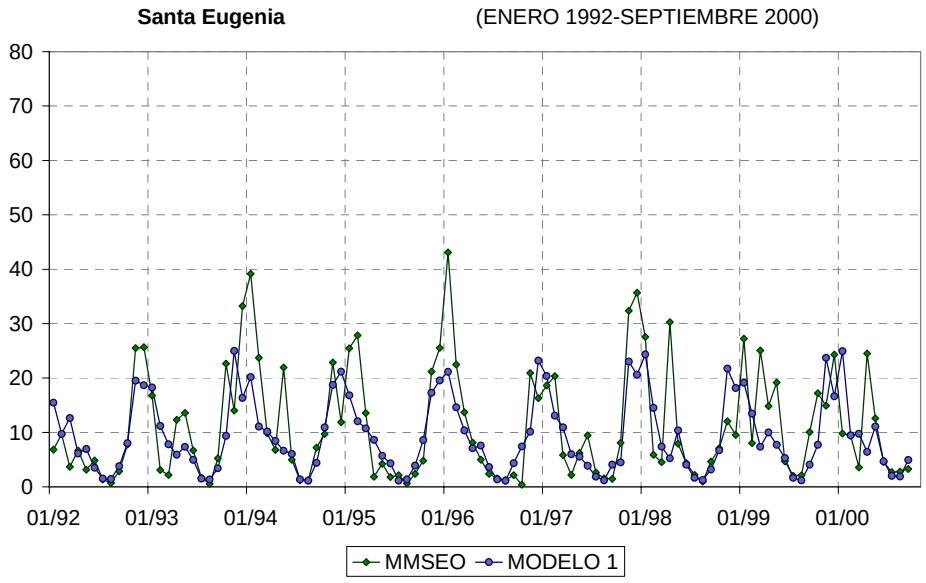

Figura 2.5. Predicción del Modelo 1 frente a la serie real. Santa Eugenia

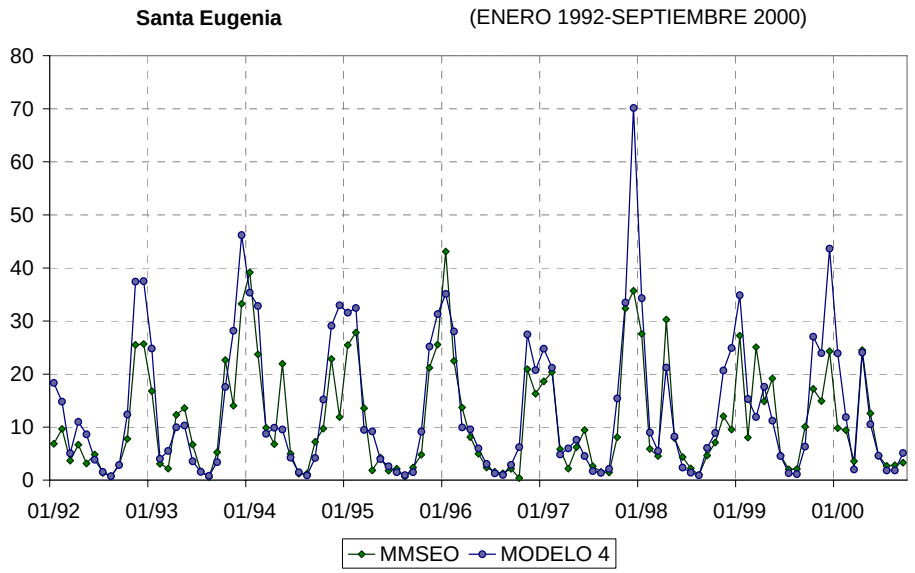

Figura 2.6. Predicción del Modelo 4 frente a la serie real. Santa Eugenia

2.4.2 Resultados Diarios

Con el fin de comparar el funcionamiento de los modelos Box-Jenkins y de las redes neuronales se han considerado dos medidas de error, la media de los valores absolutos del error relativo, *MRAE*, y el coeficiente de eficiencia, *CE. La MRAE* mide la tasa del error relativo cometido por el modelo, y el *CE* proporciona una medida de la cantidad de varianza explicada por el modelo. Se emplearon estas funciones para comparar el comportamiento de los modelos, no para entrenar las redes ni estimar los parámetros de los modelos de series de tiempo. Las expresiones explícitas responden a (2.45) para el *MRAE* y a (2.46) para *CE* (Nash y Sutcliffee, 1970)

$$MRAE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\left(Y_i - \hat{Y}_i\right)}{Y_i}\right| \tag{2.45}$$

Con la serie a estimar denominada $\{Y_i\}$ y la serie de la estimación es $\{\hat{Y}_i\}$.

$$CE = \frac{S_Y^2 - S_e^2}{S_Y^2} = 1 - \frac{S_e^2}{S_Y^2} \tag{2.46}$$

Con el error cuadrado medio, S_e^2,(2.47), eso es la variabilidad no explicada por el modelo y la varianza de la variable observada, S_Y^2,(2.48).

$$S_e^2 = \frac{\sum_{i=1}^{N} e_i^2}{N} = \frac{\sum_{i=1}^{N}\left(Y_i - \hat{Y}_i\right)^2}{N} \tag{2.47}$$

$$S_Y^2 = \frac{\sum_{i=1}^{N}\left(Y_i - \bar{Y}\right)^2}{N} \tag{2.48}$$

Se ha considerado un conjunto de validación de 77 observaciones para estimar el funcionamiento de los predictores. Tras el entrenamiento de la red neuronal se ha evaluado los errores cometido en el conjunto de validación, que fueron, por una parte, MRAE=0.178 y por otra *CE*=0.675. Estos errores significan que la tasa de error absoluto fue de 17.8%, y que el 68% de la variabilidad se explica por el modelo. Los modelo Box-Jenkins presentaron un *MRAE*=3.123, de modo que la tasa de error absoluto es del 312.2%. El coeficiente de eficiencia en el conjunto de validación fue por su parte, *CE*=-1405.845. Este resultado demuestra que los modelos Box-Jenkins no son apropiados para reproducir la claramente no lineal relación lluvia-escorrentía de la cuenca del Xallas. Las figuras 2.7 y 2.8 permiten comparar el funcionamiento de la red neuronal el modelo Box-Jenkins en el mismo período del año. La figura 2.7 muestra una sección del conjunto de entrenamiento, y la figura 2.8 corresponde al conjunto de validación.

Los resultados obtenidos por la red son prometedores, teniendo en cuanta las longitudes de las series empleadas para entrenar y validar, teniendo en cuenta además que el período de validación ha sido uno de los más lluviosos desde 1968 en Galicia, en particular desde enero de 2001, cuando comenzó la medición y el registro sistemático de los datos pluviométricos. A pesar de disponer de medidas que se remontaban a 1968 la calidad de esos años no era buena, por ausencia de largas secuencias de datos, por lo que no fueron considerados adecuados para este estudio. La bondad de los resultados contrasta con la simplicidad del modelo. Esto sustenta la idea de que en este caso concreto no era necesario realizar las medidas de los parámetros básicos de la cuenca necesarias para construir un modelo hidrológico tradicional, como temperatura, coeficiente de evapotranspiración, humedad relativa y otros parámetros asociados a la vegetación de la cuenca del río, etc. Otros trabajos confirman poca la influencia de la evaporación en modelos de predicción diarios (Danh *et al.*, 1999). Se ha, pues identificado un modelo adecuado, basado sólo en el comportamiento previo en respuesta a diferentes intensidades de lluvia. La ausencia de un modelo preestablecido de condiciones hace que los resultados obtenidos en tan simple modelo sean considerados muy satisfactorios.

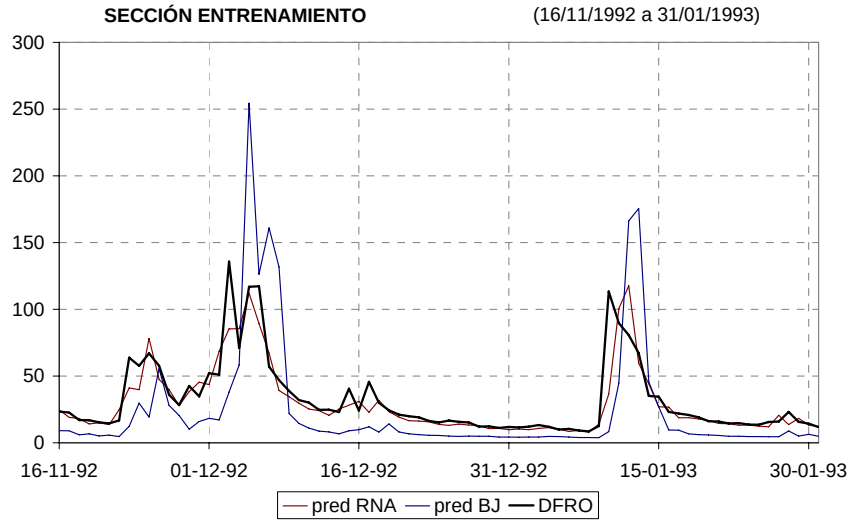

Figure 2.7. Predicciones de la red neuronal y del modelo de Box-Jenkins comparados con el valor real, sobre el conjunto de entrenamiento

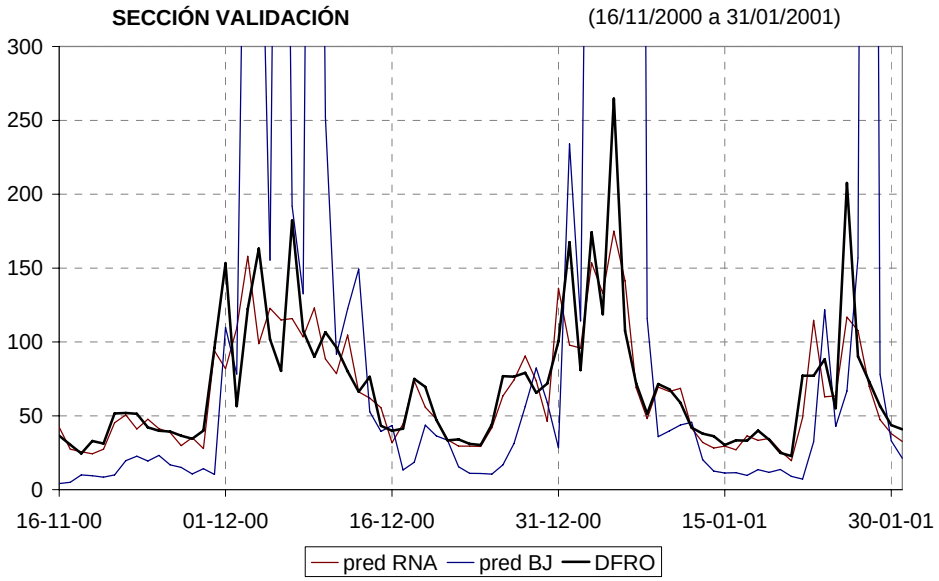

Figure 2.8. Predicciones de la red neuronal y del modelo de Box-Jenkins comparados con el valor real, sobre el conjunto de validación

En la figura 2.8 se pueden apreciar tres días (7 de diciembre, 5 y 26 de enero) en los que la predicción de la red presenta grandes diferencias con los valores observados. En estos tres días las lluvias caídas tomaron los mayores valores en décadas, de modo que en los últimos 11 años no se encuentran registros similares. Esto muestra la necesidad de disponer de un conjunto extenso de datos, para obtener buenas predicciones en cada situación. La falta de exactitud en esos días no se debe a la arquitectura de la red seleccionada, sino a las limitaciones del conjunto de entrenamiento, que en este caso, al tratarse de un método cuyo comportamiento está determinado por los datos que se le presentan. Al no haber entrenado la red con valores similares no es posible obtener una respuesta óptima en esta situación extrema. Estos días de grandes tormentas no son habituales, pero resulta evidente que son los más importantes a la hora de predecir. El funcionamiento de las redes neuronales a la predicción de picos ha sido estudiado en diversos trabajos, con resultados variables (Khalil, 2000; Maqsood *et al.*, 2002).

2.5. Conclusiones

En este capítulo se buscaba la comparación entre modelos de redes neuronales y series de tiempo a la hora de modelizar un proceso real de naturaleza continua.

Los modelos Box-Jenkins constituyen una metodología adecuada el estudio del comportamiento a largo plazo de variables hidrológicas como la lluvia o la escorrentía. Los modelos estimados proporcionan una herramienta útil para, en base a estas estimaciones tomar decisiones importante a la hora de planificar la generación de electricidad a largo plazo. Esta herramienta se implementó en MSExcel para facilitar su uso.

El otro objetivo principal en el contexto hidrográfico es mantener la seguridad de núcleos de población cercanos. El comportamiento a corto plazo de las variables hidrológicas presenta un perfil bastante errático. Los modelos lineales de series de tiempo no obtienen buenos

resultados, por lo que es importante disponer de otra herramienta para el estudio de estas variables. Las redes neuronales son modelos que permiten estimar relaciones entre variables, pues pueden ser considerados como modelos de regresión generales. En particular han sido empleados para predecir series de tiempo con éxito (Chakraborty, 1992) y se emplean cuando técnicas más clásicas no son adecuadas al problema. En este estudio se ha construido una red que reproduce el comportamiento de una sistema físico, que puede ser muy complejo de modelizar desde un a perspectiva física determinística, debido a las dificultades que entraña identificar tanto los procesos físicos que intervienen como los parámetros asociados. La red neuronal se emplea para tratar con un problema físico por lo que ha de evolucionar y variar en el tiempo, tal y como lo hace el clima y el ecosistema. Será necesario pues establecer reentrenamientos periódicos con nuevos datos, para considerar la nueva información hidrográfica disponible, así como para reflejar los posibles cambios producidos en la cuenca.

Se ha probado pues el buen funcionamiento en el contexto continuo; el siguiente capítulo se centrará en su capacidad de predecir variables binarias, esto es, en problemas de clasificación.

2.6 Bibliografía

Abbot, M.B., Bathurst, J.C., Cunge, J. A., O'Connell, P.E., Ramunsen, J. (1986) An introduction to the European Hydrological System-Systeme Hydrologique Europeen, "She": History and philosophy of a physically based distributed modelling system . Journal of Hydrology, V.87, pp. 45-59.

Abrahart, R.J., See, L., (1998) Neural Networks vs ARMA Modelling: Constructing benchmark case studies of river flow prediction.

(http://divcom.otago.ac.nz/sirc/GeoComp/GeoComp98/05/gc_05.htm)

Azoff, E.M. (1994) Neural Network Time Series Forecasting of Financial Markets. (1 Ed) John Wiley and Sons.

Bishop, C. (1995) Neural Networks for Pattern Recognition, Oxford:Clarendon Press.

Box. G.E.P. y Cox, D. R., (1964) An analysis of transformations. J. R. Statist. Soc. B, 26, pp. 211-252.

Box, G.E., Jenkins, G.M. (1976) Time series analysis: forecasting and control. Reised Edition.

Box, G.E., Jenkins, G.M., Reinsel, G.C. (1994) Time Series Analysis: Forecasting and Control. (3 Ed.) Prentice Hall.

Brockwell, P.J., Davis, R.A. (2002) Introduction to Time Series and Forecasting. (2 Ed.) Springer-Verlag.

Burke, L.I., (1991) Clustering caracterization of adaptative resonance. Neural networks, V.4(4), pp. 485-491.

Castellano-Méndez,M., González-Manteiga,W., Febrero-Bande,M., Prada-Sánchez,J.M.,Lozano-Calderón,R, (2004) Modelling of the monthly and daily behaviour of the runoff of the Xallas river using box-jenkins and neural networks methods. Journal of Hydrology, V.296, pp.38-58.

Castro, J.L., Mantas, C.J., Benítez, J.M., (2000) Neural networks, V.13, pp.561-563.

Chakraborty, K., Mehrotra, K., Mohan, C. K. (1992) Forecasting the behaviour of a multivariate time series using neural networks, Neural Networks, Vol. 5, pp. 961-970.

Chauvin, Y., y Rumelhart, D. E., (1995) Backpropagation: Theory, Architectures, and Applications. Lawrence Erlbaum Associates, Inc.

Chen, M.S, Manry, M. T., (1993) Conventional Modelling of the Multilayer Perceptron Using Polynomial Basis Functions. IEEE Trans. on Neural Net., Vol.4(1), pp.164-166.

Cheng, B., Titterington, D.M. (1994) Neural Networks a Review from Statistical Perpective. Statistical Science 9 (1), pp. 2-54

Coulibaly, P., Anctil, F., Bobée, B. (2000) Daily reservoir inflow forecasting using artificial neural networks for stopped training approach. Journal of Hydrology, V.230, pp. 244-257.

Cybenko,G., (1989) Approximations by Superpositions of a Sigmoidal Function, Math. Contr. Signals, Systems, Vol2, pp.303-314.

Dahn, N.T., Phien, H.N., Gupta, A.D. (1999) Neural network models for river flow forecasting. Water SA, V.25(1), pp. 33-39.

Deo, M.C., Thirumalaiah, K. (2000) Real time forecasting using neural networks. Artificial neural networks in Hydrology, R.S. Govindaraju, A. Ramachandra Rao (eds), Kluwer Academic Publishers, Dordrecht, pp. 53-71.

Dolling, O.R., Varas, E.A. (2002) Artificial neural networks for streamflow prediction. Journal of Hydraulic Research, V.40(5), pp. 547-554

French, M.N., Krajewski, W.F., Cuykendall, R.R. (1992) Rainfall forecasting in space and time using neural networks. Journal of Hydrology, V.137, pp. 1-31

García-Bartual, R. Short term river flood forecasting with neural networks. Available at (http://www.iemss.org/iemss2002/proceedings/pdf/volume%20due/266_bartual.pdf)

Garrick, M., Cunnane, C.,Nash J.E. (1978) A criterion of efficiency for rainfall-runoff models. Journal of Hydrology, V.36, pp. 375-381.

Goldberger,A.S. (1973) Correlations between binary choices and probabilistic predictions. Journal of the American Statistical Association, 68:84

Harmuth, H.F. (1972) Transmision of information by orthogonal functions. Springer-Verlag.

Hastie, T. (1987) A closer look at the deviance. The American Statistician, V.41(1), pp. 16-20

Hornik, K.M., Stinchcombe, M., White, H. (1989) Multilayer feedforward networks are universal approximators. Neural Networks, V.2, pp. 359 – 366

Hsu, K., Gupta, H.V., Sorooshian, S. (1995) Artificial neural networks modelling of the rainfall-runoff process. Water Resources Reesearch, V.31(10), pp. 2517-2530.

Johansson, E.M., Dowla, F.U., Goodman, D.M. (1992) Backpropagation learning for multi-layer feed-forward neural networks using the conjugate gradient method. Int. J. of Neural Systems, V.2(4), pp. 291-301.

Khalil, M., Panu, U.S., Lennox, W.C. (2001) Groups and neural networks based stream flow data infilling procedures. Journal of Hydrology, V.241, pp. 153-176.

Kneale, P.E., See, L. y Smith, A. (2001) Towards Defining Evaluation Measures for Neural Network Forecasting Models, GeoComputation 2001, 24-26 Sep 2001, Brisbane.

Leavesley, G.H., Lichty, R.W., Troutman, B.M., Saidon, L.G. (1983) Precipitation-runoff modelling system. User's manual. U.S. Geol. Surv. Water Resources Invest. Rep, pp 83-4238.

Legates, D.R. y McCabe, G.J. (1999) Evaluating the use the "goodness-of-fit" measure in hydrologic and hydroclimatic model validation. Water Resources Research, V.35, pp. 233-241.

Lekkas, D.F., Imrie, C.E., Lees, M.J. (2001) Improved non-linear transfer function and neural networks methods of flow routing for real time forecasting. Journal of Hydroinformatics, V.3(3), pp. 153-164.

Mack, Y. P. (1981) Multiple Time Series Analysis, Springer-Verlag, Heidelberg.

Maier, H.R., Dandy, G.C. (1996) The use of artificial neural networks for the prediction of water quality parameters. Water resources research, V.32, pp.1013-1022.

Makridakis, S., Wheelwright, S.C. y Hyndman, R.J. (1998) Forecasting. Methods and Applications. (3 Ed). Wiley.

Maqsood, I., Khan, M.R., Abraham, A. (2002) Neurocomputing Based Canadian Weather Analysis. Second International Workshop on Intelligent Systems Design and Application. Atlanta

Mitchell, T.M. (1997) Machine Learning. Cap 4. Artificial Neural Networks. Carnegie Mellon University Mc Graw Hill, pp. 81-127.

Nash, J.E. y Sutcliffee, J.V. (1970) River Flow Forecasting through Conceptual Models, Part 1-A discussion of principles. Journal of Hydrology, V.10, pp. 282-290.

Nor, N.I.A., Harun, S., Kassim, A.H.M. (2001) Proc. NSF Workshop, Kuala Lumpur

Park, J., y Sandberg, I.W. (1991) Universal approximation using radial basis function networks. Neural Computation, V.3, pp. 246-257.

Peña, D. (2005) Análisis de Series Temporales. Alianza Editorial.

Raman, H. y Sunilkumar, N. (1995) Multivariate modelling of water resources time series using artificial neural networks. Hydrological Sciences Journal, V.40(2), pp. 145-163.

Refsgaard, J.C. y Storm, B. (1995) MIKE SHE. In: Singh, V. J. (Ed.), Computer Models in Watershed Hydrology. Water Resour. Publications, Co., pp. 809-846.

Ripley, B.B. (1996) Pattern Recognition Using Neural Networks. Cambridge University Press.

Rosenblatt, F. (1958) The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, V.65, pp. 386-408

Rumelhart, D.E., Hinton, G.E., y Williams, R.J. (1986) Learning internal representations by error propagation. In: Rumelhart, D.E. and McClelland, J. L., eds., Parallel Distributed Processing: Explorations in the Microstructure of Cognition, V.1, pp. 318-362.

Rumelhart, D.E., Hinton,G.E., McClelland,J.L. (1996) A general framework for parallel distributed processing. In: Rumelhert, D.E. and McClelland (Ed.) Parallel Distributed Processing; Explorations in the Microstructure of Cognition, V.1, MIT Press, Cambridge.

Sajikumar, N., Thandaveswara, B.S. (1999) A non-linear rainfall-runoff model using artificial neural networks. Journal of Hydrology, V.214, pp. 32-48.

Shumway R.H., Stoffer, D.S. (2000) Time Series Analysis and Its Application. Shumway R.H., Stoffer, D.S (Ed.). Springer Text in Statistics. Springer-Verlag.

Singh,V.P., Woolhiser, D.A. (2002) Mathematical Modelling of Watershed Hydrology. Journal of Hydrologic. Engineering, V.7, pp. 270-292.

Sorooshian, S., Gupta, V.K. (1995) Model Calibration. In: Singh, V. J. (Ed.), Computer Models in Watershed Hydrology. Water Resour. Publications, Co., pp. 23-68.

Sugawara, M. (1974) Tank Model and its application to Bird Creek, Wollombi Brook, Bikin Rive, Kitsu River, Sanaga River and Namr Mune. Research note of the National Research Center for Disaster Preventions, V.11, pp. 1-64.

Sugawara, M. (1979) Automatic Calibration of the tank model. Hydrol. Sci. Bull, V.24(3), pp. 375-388.

Sugawara, M. (1995) Tank Model. In: Singh, V. J. (Ed.), Computer Models in Watershed Hydrology. Water Resour. Publications, Co., pp. 165-214.

Tang, Z., deAlmedia,C., Fishwick, P.A. (1991) Times series forecasting using neural networks vs. Box-Jenkins methodology. Simulation, V.57, pp. 303-310.

Uvo, C.B., Tolle, U., Berndtsson, R. (2000) Forecasting discharge in Amazonia using artificial neural networks. Int. J. Climat. V.20, pp.1495-1507.

Wei, W.W. (1990) Time Series Analysis. Univariate and Multivariate Methods. Addison-Wesley.

Whitehead, B, Choate, T. (1996) Cooperative-competitive genetic evolution of Radial Basis Function centers and widths for time series prediction. IEEE Trans. on Neural Networks,V.7(4), pp. 869-880.

Widrow, B., Hoff, M.E. (1960) Adaptive switching circuits. WESCON. Conv. Record, part IV, pp. 96-104.

Zealand, C.M., Burn, D.H., Simonovic, S.P. (1999) Short term streamflow forecasting using artificial neural networks. Journal of Hydrology, V.214, pp. 32-48.

Zhao, R. J., Zhang, Y. L., Fang, L. R., Xiu, X. R., Zhang, Q. S. (1980) The Xinanjiang model. In "Hydrological Forecasting", Proceed. Oxford Symposium, IAHS Publ. V.129, pp. 351-356.

Capítulo 3. Redes Neuronales en Problemas de Clasificación.

RESUMEN

En este capítulo se abordará la aplicación de las redes neuronales a problemas de clasificación, en particular a problemas en los que la variable respuesta es binaria. En general se han aplicado redes de base radial para estas tareas; en este estudio de muestra que los perceptrones son también adecuados para resolver problemas de este ámbito. Se han realizado dos estudios, uno aplicado a datos reales en el que se predicen los niveles de alerta de riesgo alergénico causado por presencia de polen de *Betula* y un estudio de simulación en el que se compara la actuación de modelos lineales generalizados frente a redes neuronales bajo distintos escenarios. En ambos casos los resultados han sido muy positivos para las redes neuronales y si aplicación a problemas de clasificación en los que el modelo subyacente es, o bien complejo, o desconocido.

Parte de los resultados que se detallan en este capítulo están recogidos en el artículo de Castellano-Méndez *et al*, 2005.

3.1 Aplicación a las Ciencias Medioambientales. Predicción de Niveles de Riesgo de Polen de Betula en el Aire

3.1.1 Introducción al Problema

Un porcentaje creciente de la población europea sufre de alergias al polen. El estudio de la evolución de concentración de polen de aire puede proporcionar información sobre los niveles previos de polen en el aire, lo que puede ser útil para la prevención y el tratamiento de síntomas alérgicos, y la administración de recursos médicos. Los síntomas de polinosis por Betula pueden ser asociados a ciertos niveles de polen en el aire. El objetivo de este estudio es predecir el riesgo de que la concentración de polen exceda un nivel dado, usando el polen anterior y la información meteorológica aplicando técnicas de redes neuronales. Las redes neuronales son un instrumento extendido estadístico útil para el estudio de problemas asociados con el complejo o fenómenos mal entendidos. La variable de respuesta binaria asociada con cada nivel requiere una selección cuidadosa de la red neuronal y la función de error asociada con el algoritmo de aprendizaje usado durante la fase de entrenamiento. El funcionamiento de la red neuronal en el conjunto de validación muestra que el riesgo de que el nivel de polen que exceda un cierto umbral puede ser pronosticado con éxito usando redes neuronales artificiales. Este instrumento de predicción puede ser puesto en práctica para crear un sistema automático que pronostique el riesgo de sufrir síntomas alérgicos.

El abedul es un árbol anemófilo con la alta producción de polen (Moore y Webb, 1978; Lewis *et al.* 1983), cuya capacidad alergénica ha sido citada por numerosos autores (Spieksma, 1990; Norris-Hill y Emberlin, 1991; D'Amato y Spieksma, 1992). Su polen se considera la causa principal de polinosis en el norte y centro de Europa (Wihl *et al.*, 1998; Spieksma *et al.*, 1995) no sólo durante su estación de polen sino también durante períodos anteriores y subsecuentes, pues su polen fácilmente puede ser transportado a lo largo de largas distancias (Wallin *et al.*, 1991; Hjelmroos, 1991). En tales casos, la actividad antígena parece estar vinculada a alérgenos depositados sobre partículas de polvo dentro de las casas, una característica de los granos de polen de abedul, que pueden provocar el inicio de procesos alérgicos incluso hasta dos meses después de que las concentraciones de polen máximas en el aire tuviesen lugar (Ekebom *et al.*, 1996; Rantio-Lehtimaki *et al.*, 1996). El predominio de polen de abedul alcanza el 13 % al 60 % en la población afectada de polinosis en las mismas localidades en Galicia-N.W. España - (Arenas *et al.*, 1996; Aira *et al.*, 2001) y el 19 % en Santiago de Compostela (Dopazo, 2001).

Diversos investigadores han realizado estudios aeropalinológicos sobre esta planta, para determinar el modelo del comportamiento estacional y diario de polen de abedul y la influencia de los parámetros diferentes meteorológicos sobre la concentración de polen (Spieksma *et al.*, 1989; Atkinson y Larsson, 1990; Norris-Hill y Emberlin, 1991; Spieksma *et al.*, 1995; Aira *et al.*, 1998; Jato *et al.*, 2000; Latalowa *et al.* 2002). De este modo, se pueden establecer modelos para predecir tanto el comienzo como la severidad de la estación de polen. Se emplearon diferentes factores como predictores del principio de la estación de polen en los diferentes modelos. Por ejemplo la suma de temperaturas hasta una fecha fue utilizada por el Coágulo (2001), Caramiello *et al.* (1994), Ruffaldi y Greffier (1991). En otros trabajos se emplearon factores fenológicos como unidades de frío y días del grado de crecimiento como predicotres (Andersen, 1991). Larsson (1993) empleó el método de actividad acumulada y Laaidi (2001) empleó conjuntamente la suma de las temperaturas y un modelo de regresión múltiple. Se realizaron diferentes trabajos con el objetivo de construir modelos para predecir la concentración de polen media con un día de anticipación empleando regresión lineal (Rodríguez-Rajo, 2000; Méndez, 2000) o series de tiempo (Moseholm *et al.*, 1987). Sin embargo no hay trabajos para conocer el riesgo de la cantidad de granos de polen en el aire supere un umbral.

Los síntomas de la polinosis por Betula pueden para aparecer provocados por diferentes niveles de polen Betula en el aire, dependiendo de las diferentes características de cada individuo. Sin embargo varios valores umbrales han sido establecidos como valores límite para la aparición de síntomas. En diversos estudios el 90 % de los sujetos clínicamente sensibles muestran síntomas cuando se alcanzan los 30 polen granos/m^3 y el inicio de síntomas severos se produce con concentraciones superiores a 30 polen granos/m^3 (Viander y Koivikko en Negrini *et al.*, 1992). Corsico (1993) consideró el mismo nivel como el umbral para el principio de los síntomas alérgicos. El polen de abedul es muy abundante en el aire de Santiago de Compostela en marzo y abril y concentraciones superiores a 100 granos/m^3 son frecuentes. Los niveles diarios máximos se registran por la tarde, entre las 12 y las 18 horas y coinciden con el momento de mayor frecuencia de síntomas de alergia (Dopazo, 2001)

El abedul está representado en Galicia por una especie, Betula Alba L. (Moreno, 1990). Esta especie está extensamente distribuida en nuestra área y conforma como el árbol dominante, los bosques altimontanos oro-Cantábricos acidófilos, con una distribución claramente Euro-siberiana. Se encuentran en altitudes superiores a 1,150 metros, siendo las últimas formaciones arbóreas de la secuencia altitudinal, con termo-climas de montaña y ombro-climas hiperhúmedos (Izco, 1994). Su límite, aunque claramente polémico, se sitúa en las sierras gallegas de Ancares y Caurel (Costa *et al.,* 1990). En esta misma área, pero sobre suelos silíceos y con una influencia mediterránea mayor, hay también bosques de abedules en la capa de altimontaña Galaicoortuguésa y en la zona supra-Mediterránea de Ourense-Sanabrian.

En las montañas y colinas de Galicia se pueden encontrar abedules no climáticos, en sustitución de arboledas de roble, que se localizan sobre suelos ácidos y con límites altitudinales entre 600 y 1.100 metros.

En la región Euro-siberiana, el abedul puede formar la parte de bosques ribereños, con *Alnus glutinosa*, *Salix atrocinera* y *Frangula alnus*. Se puede encontrar *Betula* como árboles ornamentales, y este es uno de los motivos por los que las concentraciones de polen Betula alcanzan en Santiago los valores más altos en Galicia.

El objetivo de esta investigación es la detección de días de alto riesgo alérgico durante la polinización de *Betula*, usando redes neuronales artificiales, con el fin de alertar tanto al especialista en alergias como a la población con problemas alérgicos de una situación de riesgo potencial. Las redes neuronales artificiales, tal y como se expone en el capítulo 1, son herramientas estadísticas completas para el análisis de datos (Bishop, 1995). Extendido su uso en muchos ámbitos, las redes neuronales han sido también empleadas en estudios aerobiológicos, para obtener modelos de predicción capaces de mejorar el pronóstico de la concentración de polen diaria (Ranzi, 2000; Hidalgo *et al.*, 2002; Duna de Sánchez *et al.*, 2002)

3.1.2 Material y Métodos

El estudio fue realizado en la ciudad de Santiago de Compostela, situada en el noroeste España, como muestra la Figura 3.1. La monitorización del polen fue realizada desde 1993 hasta el 2001 mediante un muestreador de aire volumétrico de 7 días (Lanzoni VPPS, 2000) situado aproximadamente 25 metros por encima del nivel de tierra. La metodología empleada para tratar e interpretar las muestras fue la recomendada por la Red Española de Aerobiología, REA, (Dominguez, 1995).

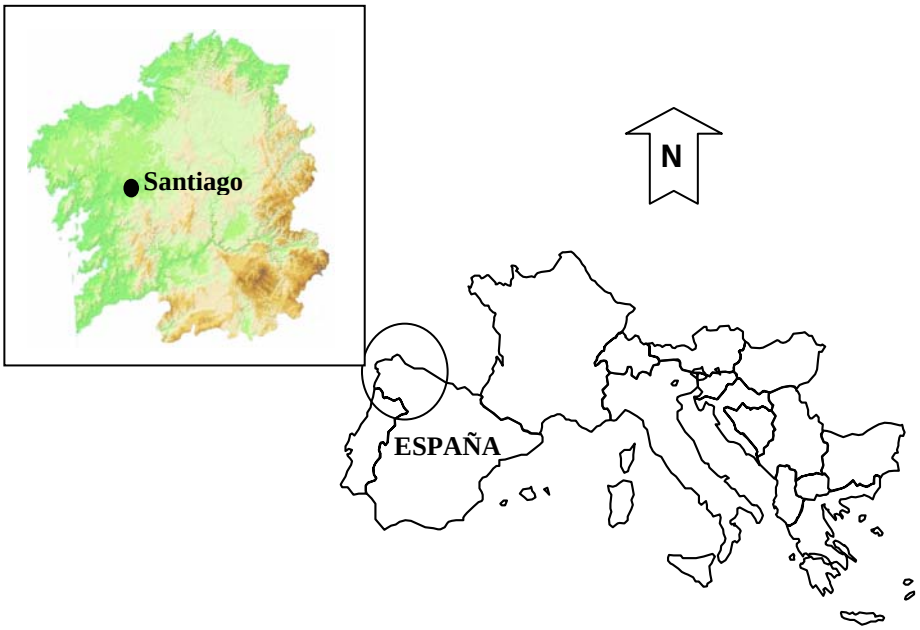

Figura 3.1. Localización de Santiago de Compostela en Europa

Se han considerado tres series de datos (Chakraborty, 1992). El polen diario, $pollen_t$, expresado como granos/m^3, y dos series exógenas meteorológicas, la lluvia del día, DR_t, expresada como l/m^2, y la media diaria de la temperatura, DMT_t, expresada como grados centigrados (ºC).

El objetivo de este trabajo no es predecir la concentración de polen, sino la existencia de un alto riesgo alérgico. Considerando un nivel de polen dado, *lev*, se puede definir una variable binaria Y_t, de modo que tome el valor 1 si $pollen_t$ es superior a esa cantidad, *lev*, y 0 si no lo es. Los niveles seleccionados han sido 20, 30, 70 y 80 granos/m^3. Para niveles 20 y 30 la variable Y_t mide el riesgo de que se produzcan síntomas alérgicos iniciales y para 70 y 80 mide el riesgo de síntomas severos para el 90% de la población más alérgica. Las variables dependientes u objetivo serán los Y_t asociados a cada nivel.

Las variables seleccionadas independientes han sido las precipitaciones del día anterior, DR_{t-1}, la temperatura media del día anterior, DMT_{t-1}, y la concentración de polen del día previo, $pollen_{t-1}$.

El método estadístico usado para el estudio y el pronóstico del nivel de riesgo de polen de *Betula* es la técnica de redes neuronales artificial (Ripley, 1996). Las redes neuronales son métodos en los que son los datos los que hablan, i.e. los que determinan la estructura de la red, por lo que la relación entre las entradas y las salidas depende de un conjunto histórico de observaciones llamado conjunto de entrenamiento, usado para el estudio de red. Como se comenta en la introducción le conjunto de entrenamiento es una colección de datos relacionada con situaciones pasadas y asociado a ellos, la respuesta deseada de la red neuronal o cierta variable estrechamente relacionada con la respuesta correcta, que es desconocida.

El algoritmo de entrenamiento empleado ha sido el algoritmo de backpropagation (Rumelhart *et al.,* 1986; Chauvin *et al.*, 1995). Después de la fase de entrenamiento, cuando la red actúia ante una nueva situación, lo hará de modo coherente con lo aprendido. Las redes tienen

interés para la predicción de datos procedentes de procesos desconocidos o complejos. La dispersión del polen es un problema muy complejo que involucra una gran cantidad de información meteorológica (la dirección de viento, la velocidad de viento, la lluvia...), ecológica (la situación forestal y la concentración de la especie seleccionada en las proximidades de la posición de predicción, ...), y topográfica (colinas, valles, ríos, ciudades, posición exacta), que no siempre está disponible. Las redes son un instrumento útil porque no requieren de la determinación de todas estas características. En cambio pueden suplirse por una estructura general de red y un conjunto de datos diverso y extenso que se pueda emplear en el entrenamiento. En la introducción ya se ha hablado considerablemente de las redes neuronales

Como la relación entre variables de entrada y salida o variables objetivo, en este caso las características meteorológicas o ecológicas del área, pueden sufrir variaciones en el futuro, resulta útil reciclar a red periódicamente, ampliando o modificando el conjunto de entrenamiento con nuevos datos que reflejen los cambios de las variables en el tiempo.

Una de las arquitecturas de Ann más populares es el perceptrón multicapa, ya presentado ampliamente en el capítulo 1. Esta ha sido la arquitectura seleccionada para abordar este problema. La figura 3.2 vuelve a mostrar su topología.

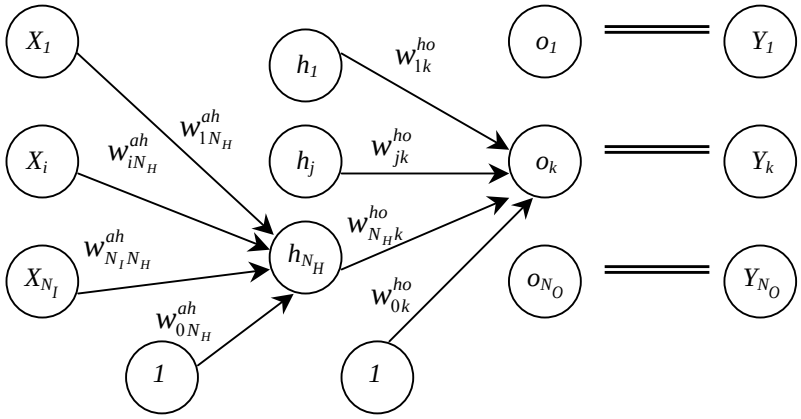

Figura 3.2. Perceptrón Multicapa con una capa oculta (N_I-N_H-N_O) con N_I variables de entrada X_i, N_H nodos en la capa oculta, N_O predicciones o_k y variables objetivo Y_k

3.1.2.1 Redes Neuronales para datos con Respuesta Binaria.

La variable objetivo en este problema, Y_t , es una variable binaria, esto es, esto toma sólo dos valores, uno o cero, es necesario, pues, seleccionar un topología de redes adecuada a esta característica. En este caso el estudio se ha centrado en una familia especial de redes, aquellas con nodos en la capa de salida sigmoideos, apropiada para el procesamiento de daros con variable respuesta binaria.

En los problemas de predicción el objetivo es acercarse sobre el valor esperado de la variable objetivo condicionada por las variables independientes. Para variables objetivo binarias esta esperanza es la probabilidad de que Y_t tome valor uno, condicionado por los valores de variables de entrada (Goldberger, 1973; Agresti, 1990). Esta probabilidad condicional puede ser consideran como una función desconocida de las variables independientes, que toma

valores del cero a uno (McCullagh y Nelder, 1989). La estimación de esta función será el objetivo de la red neuronal.

Dado un predictor, es posible construir una familia de clasificadores $F = \{C_p / p \in [0,1]\}$. Cada uno de estos clasificadores, determinados por un *p* dado, permite construyen una variable binaria a partir del predictor mediante el procedimiento siguiente. Si la probabilidad predicha es menos que *p* el valor predicho para Y_t será el cero, y en otro caso será uno. Para obtener una estimación Y_t, es necesario seleccionar un clasificador de esta familia, escogiendo un umbral *p*. En este caso el *p* seleccionado fue 0.5.

Usando la predicción de probabilidad de la red, la predicción de Y_t se obtiene de modo que si la probabilidad predicha es menos de 0.5 la estimación de Y_t será el cero, y en otro caso será uno.

3.1.2.2 Función de Error para Variables Objetivo Binarias.

La valoración de funciones de densidad, así como la valoración de probabilidades es un problema de aprendizaje sin supervisión, por refuerzo. Tanto las verdaderas probabilidades como la verdadera densidad no son conocidas en ningún caso; en cambio se dispone de una variable binaria, *Y*, que proporciona cierta información indirecta sobre el valor de la probabilidad.

En la introducción se explicó detalladamente que el funcionamiento de las redes neuronales está basado en un algoritmo de entrenamiento. Esta clase de algoritmo se compara la salida obtenida con la verdadera variable objetivo, de modo que modifica los parámetros de red para reducir al mínimo las diferencias entre ellos. Estas diferencias se evalúan a través de una función de error. La selección de una función de error apropiada para los datos resulta esencial para entrenar la red de modo satisfactorio. En problemas con variables dependientes binarias el algoritmo de entrenamiento comparara la variable binaria objetivo con la salida continua de la red, que estima la probabilidad de que la variable objetivo tome el valor 1.

Cuando se trabaja con variables objetivo binarias, la función de error habitual es la *desviance* (Hastie, 1987), *dev*, que mide en cierta manera la credibilidad de la estimación de probabilidad, dado el valor de la variable binaria. Si se denota por *o* la salida de la red, y por *Y* la variable objetivo la deviance responde a (3.1) para una muestra de tamaño 1, y para una muestra de tamaño n.

$$Dev(o,Y) = -2\left[YLog(o) + (1-Y)\,Log(1-o)\right] \tag{3.1}$$

$$Dev(\hat{P},Y) = -2\left[\sum_{i=1}^{n} Y_i Log(\hat{P}_i) + (1-Y_i)\log(1-\hat{P}_i)\right] \tag{3.2}$$

Como se comentó anteriormente la predicción de la probabilidad proporciona una variable binaria, $\hat{Y}$, que estima la variable objetivo binaria *Y*. Este es un problema de clasificación de dos clases, luego la probabilidad de clasificación incorrecta, *mcp*, puede ser considerada como la función de error; la *mcp* es la probabilidad de tener una variable objetivo que tome

valor 0 y la estimación sea 1, MC_I, más la probabilidad de tener una variable objetivo con valor 1, y que la estimación sea 0, MC_{II} (3.3).

$$mcp = P\left(\hat{Y} \neq Y\right) = MC_{I} + MC_{II} = P\left(\hat{Y} = 1, Y = 0\right) + P\left(\hat{Y} = 0, Y = 1\right) \tag{3.3}$$

$$mcp = P\left(Y = 0\right) \cdot P\left(\hat{Y} = 1 \middle| \; Y = 0\right) + P\left(Y = 1\right) \cdot P\left(\hat{Y} = 0 \middle| \; Y = 1\right) \tag{3.4}$$

$$error_{I} = P\left(\hat{Y} = 0 \middle| \; Y = 1\right) \tag{3.5}$$

$$error_{II} = P\left(\hat{Y} = 1 \middle| \; Y = 0\right) \tag{3.6}$$

$$mcp = P\left(Y = 1\right) \cdot error_{I} + P\left(Y = 0\right) \cdot error_{II} \tag{3.7}$$

$$P\left(A\right) = \frac{\text{Number of cases where } A \text{ ocurrs}}{\text{Total number of cases}} = \frac{card\left(\{A \text{ ocurrs}\}\right)}{card\left(\{\text{all examples}\}\right)} \tag{3.8}$$

Se pueden considerar de modo separado el error tipo I(3.5), $error_I$, y el error tipo II (3.6), $error_{II}$. El error tipo I es la probabilidad de que la predicción sea 0 condicionado a que la variable objetivo tome valor 1; en este problema equivale a la probabilidad de predecir que el nivel de polen estará bajo un umbral cuando en ese día el nivel esté sobre el umbral (falsa seguridad). El error de tipo II, la probabilidad de estimar 1, condicionado para a que la variable objetivo valga 0, en este caso es la probabilidad de predecir un día con un nivel de polen sobre el umbral durante un día de nivel bajo umbral (falsa alarma). Es necesario un equilibrio entre ambos errores. Por lo general se fija uno de los errores, y se selecciona el clasificador que reduce al mínimo el otro error. Todas estas medidas de error se estiman a partir de probabilidades empíricas (3.8), sobre una colección de observaciones, F.

En muchos problemas reales, por ejemplo, ecológicos, epidemiológicos o problemas médicos, no se le asigna la misma importancia a los dos tipos de error, de modo que uno de los tipos ha de ser penalizado. Una de las posibles causas es que los dos posibles valores de la variable respuesta no se presenten en la misma proporción. En muchos problemas las consecuencias económicas o para la salud de los falsos negativo son muy diferente de las consecuencias de un falso positivo, por lo que es necesario penalizar de modo diferente ambos errores.

La deviance no distingue entre ambos tipos de error, lo que lleva a la red neuronal a alcanzar un equilibrio paritario entre ambos errores empíricos. En este problema, la proporción de días que toman valor 1 es muy pequeña comparada a la proporción de días con valor 0, por lo que la desviación no es la mejor opción de función de error, por lo que se ha considerado otra función de pérdida. Se ha definido una nueva función de error (3.9), en la que se denota por *o* la salida de la red, esto es, la probabilidad estimada, y por *Y* la variable objetivo binaria.

$$error_{K_1 K_2} = K_1 Y\left(1 - o\right) + K_2\left(1 - Y\right)o, \text{ con } K_1, K_2 > 0 \tag{3.9}$$

Esta función penaliza ambos errores de modo independiente. El valor de las constantes K_1 y K_2 determinará qué error es más importante. Durante el entrenamiento está será la función que se intente minimizar, o de modo equivalente:

$$error_K = KY(1-o) + (1-Y)o, \text{ con } K > 0 \quad (3.10)$$

El valor de la constante *K* determina el penalización. Si *K*>1 el error de clasificación tipo I será el más penalizado, mientras que si *K* <1 lo será el de tipo II. Finalmente si *K*=1 ambos errores de clasificación será considerado como igualmente graves.

3.1.3 Resultados y Discusión

Se consideraron cuatro niveles de concentración de polen, lev = 20, 30, 70, 80 granos/m^3. De sete modo se han construido cuatro redes neuronales artificiales, un para cada nivel de alerta. Las redes empleadas fueron perceptrones con 5 nodos en la capa oculta. Las ecuaciones (3.11) a (3.14) muestran las expresiones explícitas de cada red, mientras que la Tabla 3.2 muestra los valores de los parámetros para cada red, según el esquema de la Tabla 3.1.

$$\hat{Y}_t = threshold(o_t - 0.5) \quad (3.11)$$

$$o_t = sigmoid\left(-w_{01}^{ho} + w_{11}^{ho} \cdot h_1 + w_{21}^{ho} \cdot h_2 - w_{31}^{ho} \cdot h_3 + w_{41}^{ho} \cdot h_4 - w_{51}^{ho} \cdot h_5\right) \quad (3.12)$$

$$h_j = sigmoid\left(w_{0j}^{ah} - w_{1j}^{ah} \cdot DR_{t-1} - w_{2j}^{ah} \cdot DMT_{t-1} + w_{3j}^{ah} \cdot pollen_{t-1}\right), \text{ with } 1 \le j \le 5 \quad (3.13)$$

$$sigmoid(x) = \frac{\exp(x)}{1+\exp(x)} \quad threshold(x) = 1 \text{ if } x \ge 0 \text{ and } 0 \text{ if } x < 0 \quad (3.14)$$

MATRICES DE PARÁMETROS

$$\begin{pmatrix} w_{01}^{ah} & w_{02}^{ah} & w_{03}^{ah} & w_{04}^{ah} & w_{05}^{ah} \\ w_{11}^{ah} & w_{12}^{ah} & w_{13}^{ah} & w_{14}^{ah} & w_{15}^{ah} \\ w_{21}^{ah} & w_{22}^{ah} & w_{23}^{ah} & w_{24}^{ah} & w_{25}^{ah} \\ w_{31}^{ah} & w_{32}^{ah} & w_{33}^{ah} & w_{34}^{ah} & w_{35}^{ah} \end{pmatrix}$$

$$\begin{pmatrix} w_{01}^{ho} & w_{11}^{ho} & w_{21}^{ho} & w_{31}^{ho} & w_{41}^{ho} & w_{51}^{ho} \end{pmatrix}$$

Tabla 3.1 Estructura de las matrices de parámetros

El período entre el 2 de enero de 1993 y el 11 de marzo de 2000, es el conjunto de datos empleado para entrenar las redes neuronales. Para evaluar el funcionamiento de las mismas ante de una nueva situación se ha considerado un conjunto de validación que comprende del 12 de marzo de 2000 al 1 de diciembre de 2001.

NIVEL 20	NIVEL 70
$\begin{pmatrix} -0.752 & 1.430 & -1.375 & -1.846 & -1.041 \\ -1.643 & 0.657 & -1.282 & 1.685 & 0.158 \\ -0.391 & -1.022 & 0.335 & -0.250 & 1.530 \\ 1.938 & 0.530 & -0.545 & 0.858 & 0.174 \end{pmatrix}$ $\begin{pmatrix} -0.109 & 0.748 & 0.082 & -0.145 & 0.045 & -0.346 \end{pmatrix}$	$\begin{pmatrix} -1.323 & 0.033 & 1.817 & -0.675 & 1.359 \\ 0.006 & 0.789 & -0.082 & -0.543 & 0.024 \\ -0.804 & -2.078 & 0.804 & -1.706 & -0.861 \\ -1.931 & 0.730 & -1.171 & 1.900 & 0.900 \end{pmatrix}$ $\begin{pmatrix} -1.547 & 0.521 & 0.794 & -1.155 & 0.650 & 0.364 \end{pmatrix}$
NIVEL 30	**NIVEL 80**
$\begin{pmatrix} 1.275 & 1.129 & 1.769 & 1.149 & 0.034 \\ 1.883 & 1.398 & -2.139 & 1.853 & 0.007 \\ 1.051 & -0.734 & 1.298 & 0.328 & -1.081 \\ -1.482 & 1.436 & 0.096 & 0.533 & 1.812 \end{pmatrix}$ $\begin{pmatrix} -0.355 & -0.306 & 0.858 & -0.103 & -0.652 & 0.880 \end{pmatrix}$	$\begin{pmatrix} 0.411 & -0.379 & 0.017 & 1.221 & 1.040 \\ 0.352 & 0.086 & 1.013 & -0.863 & -0.150 \\ -1.677 & -1.158 & -1.745 & 2.001 & -0.906 \\ 0.663 & 1.527 & 1.081 & -0.576 & 1.509 \end{pmatrix}$ $\begin{pmatrix} -1.179 & 0.246 & 0.370 & 0.403 & 0.866 & 0.560 \end{pmatrix}$

Tabla 3.2 Valores de la pesos de las Redes para cada nivel de alerta

La validación seleccionada contiene dos períodos de polinización de *Betula* consecutivos, debido al comportamiento bianual del polen estudiado.

El parámetro la *K* involucrado en la función de error toma valores diferentes para los niveles diferentes. Los valores de *K* han sido seleccionados de modo que tome valores próximos a la proporción empírica entre los días con nivel de polen bajo el umbral y los días de niveles de polen sobre el umbral, Z_{lev}, (3.15) De hecho, si la *K* escogida es Z_{lev} la red reduce al mínimo la probabilidad de clasificación incorrecta.

$$Z_{lev} = \frac{card\left\{t / Y_t = 0\right\}}{card\left\{t / Y_t = 1\right\}} \tag{3.15}$$

Durante la polinización de *Betula* el número de días con polen en el aire es menor que el número de días en los que la concentración de polen es el cero, por lo que *K* es mayor que uno. Los valores más altos de *lev* están asociados a valores de Z_{lev} más altos, y por tanto valores de *K* más altos.

Para mostrar, comparar y discutir los resultados obtenidos es necesario definir ciertas medidas de comparación. Se considerarán dos medidas de concordancia diferentes y complementarias. La proporción de clasificación correcta sobre el conjunto de observaciones en las la variable objetivo vale 1 (sobre el nivel), GC_I,(3.16) y la proporción de clasificación correcta sobre el conjunto de observaciones en las la variable objetivo vale 0 (bajo el nivel), GC_{II}, (3.17).

$$GC_I = 1 - \hat{P}\left(\hat{Y} = 0 \middle| Y = 1\right) = \hat{P}\left(\hat{Y} = 1 \middle| Y = 1\right) = \frac{card\left\{t / Y_t = \hat{Y}_t = 1\right\}}{card\left\{t / Y_t = 1\right\}} \tag{3.16}$$

$$GC_{II} = 1 - \hat{P}\left(\hat{Y} = 1 \middle| Y = 0\right) = \hat{P}\left(\hat{Y} = 0 \middle| Y = 0\right) = \frac{card\left\{t / Y_t = \hat{Y}_t = 0\right\}}{card\left\{t / Y_t = 0\right\}} \tag{3.17}$$

La tabla Tabla 3.3 muestra los resultados obtenidos.

La figura 3.3 muestra la predicción de las probabilidades condicionadas junto a la variable objetivo, para los niveles 30 y 80 sobre una sección del conjunto de entrenamiento y otra sección del conjunto de validación. La línea punteada horizontal separa las dos zonas de predicción. Si la estimación de la probabilidad condicional está sobre la línea la predicción binaria tomará el valor 1, y si está bajo la línea de puntos, la predicción binaria tomará el valor 0.

		$nivel = 20$	$nivel = 20$	$nivel = 20$	$nivel = 20$
GC_I	ENTRENAMIENTO 1993-1999	0.95	0.95	0.86	0.86
	VALIDACIÓN 2000	0.92	0.88	0.83	0.83
	VALIDACIÓN 2001	1.00	1.00	1.00	1.00
GC_{II}	ENTRENAMIENT 1993-1999	0.95	0.95	0.98	0.98
	VALIDACIÓN 2000	0.93	0.92	0.92	0.93
	VALIDACIÓN 2001	0.96	0.97	0.97	0.97

Tabla 3.3 Resultados obtenidos en términos de clasificación correcta

La predicción del comportamiento del polen se ha convertido en un objetivo importante dentro de la aerobiología. El objetivo es proporcionar información fiable y exacta sobre el polen presente en el aire a aquellos usuarios que son sensibles, con el fin de ayudarles a optimizar su medicación.

Por lo general, la divulgación de la información aerobiológica se realiza empleando categorías fijas relacionadas con ciertos valores umbral. En este sentido, el objetivo en este trabajo era aplicar modelos de redes neuronales para estimar la probabilidad de que la concentración diaria de polen *Betula* supere ciertos umbrales, algunos de cual antes han sido citados como responsables de aparición de síntomas alérgicos (Negrini *et al.*, 1992; Corsico, 1993).

Las redes neuronales proporcionaron resultados satisfactorios a la hora de pronosticar la probabilidad de que se supere que un valor dado de concentración de polen *Betula*. Entre el 83 y el 92% de los episodios en el año 2000, y el 100% de los episodios en el año 2001, en que los valores de concentración de polen alcanron los umbrales considerados (> 20,> 30,> 70 y> 80 granos/m^3) fueron predichos con éxito y con anticipación. Asimismo en el año 2000, se predijeron entre el entre 92 y el 93% de los días con niveles de polen bajo los umbrales, mientras que en el 2001 los porcentajes de predicción correcta se situaron según los diferentes niveles entre el 96 y el 97 %.

Por lo tanto, las redes neuronales son un instrumento adecuado y útil a la hora de predecir la probabilidad de exceder un cierto valor de umbral, y de este modo son útiles a la hora de divulgar la información aerobiológica entre la población que sufre de problemas alérgicos. Este es un primer paso para la automatización de un sistema de predicción y alerta de episodios alérgicos.

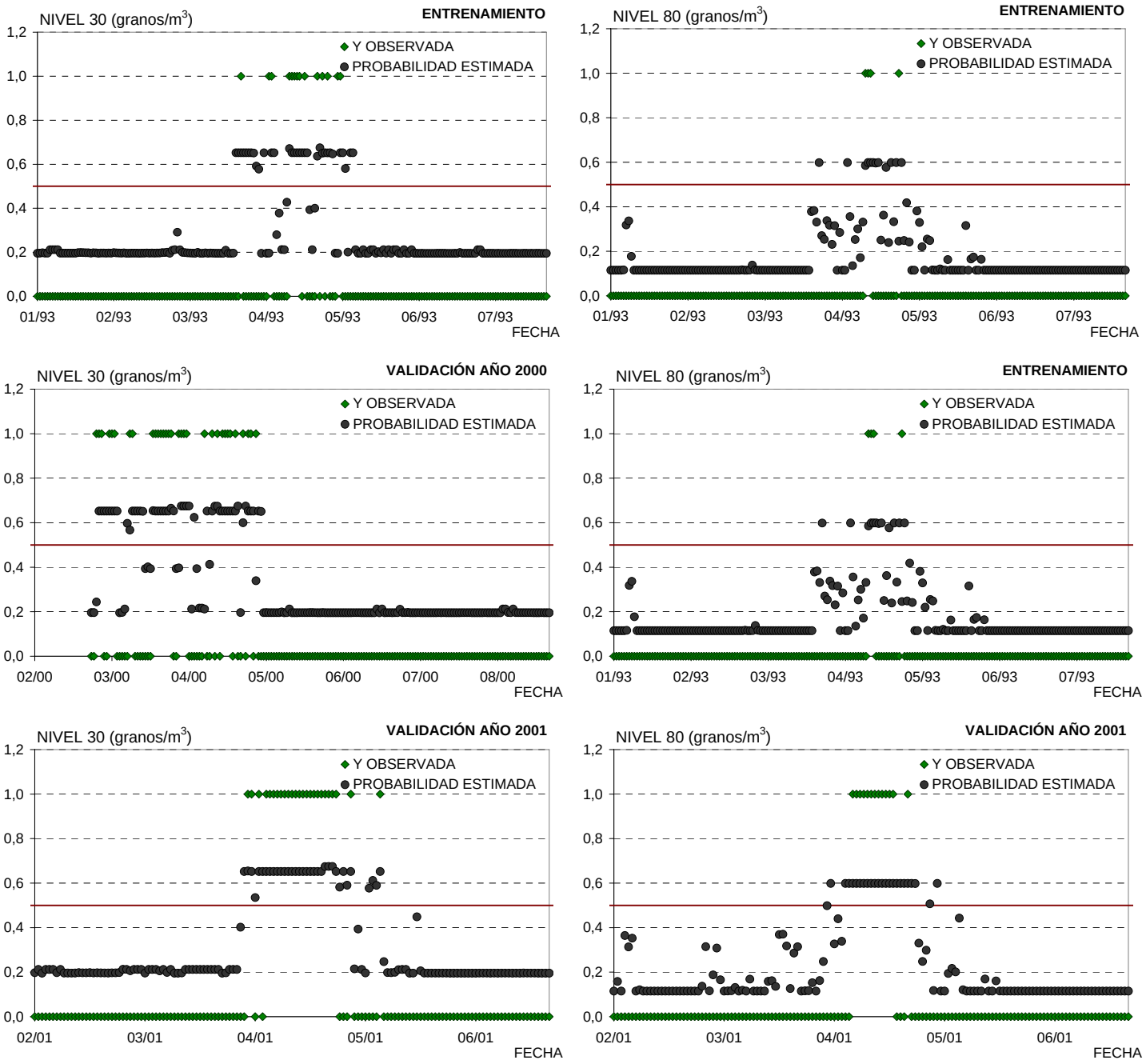

Figura 3.3 Predicción de probabilidades condicionadas junto a la variable objetivo. Niveles 30 y 80 granos/m^3.

3.2 Aplicación a un problema de simulación. Comparación de los Modelos Lineales Generales y las Redes Neuronales.

3.2.1 Introducción

Con el fin de evaluar de modo objetivo la competitividad del funcionamiento de las redes neuronales resulta interesante comparar el funcionamiento de una red neuronal con algún otro método estadístico, en particular, en problemas de predicción binaria el candidato adecuado

para la comparación puede ser un modelo de regresión lineal generalizado. Para ello se emplearán datos simulados, en distintos escenarios.

3.2.2 Modelo Lineal Generalizado

En el capítulo 1 se introdujo brevemente lo que constituye un modelo lineal generalizado. En esta sección se detallará más la idea que lo sustenta y la metodología de estimación.

En un contexto de teoría de la regresión asociada a un vector (Z,Y), sea *Z* el vector regresor p-dimensional e *Y* la variable respuesta unidimensional. Los modelos de regresión generalizados se componen de tres componentes:

La *componente sistemática* $\eta = \eta(\mathrm{z})$ que es una transformación unidimensional del vector de covariables; *la componente aleatoria* que consiste en suponer que la distribución de la variable *Y* dado *Z* pertenece a la familia exponencial, y finalmente *la función link* $H(\cdot)$ que relaciona las dos componentes anteriores a través de la expresión (3.18).

$$E\left[Y\middle|Z\right] = H\left(\eta\right) \tag{3.18}$$

Como caso particular en los modelos lineales generalizados (GLM) se supone que la componente sistemática es una combinación lineal y aditiva de las covariables, es decir, (3.19) donde β es un vector de parámetros desconocido.

$$\eta = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + \ldots + \beta_p z_p = \beta^{\mathrm{T}} \mathrm{Z} \tag{3.19}$$

Por ejemplo, si la variable es binaria y la función link es la función logit el modelo que resulta responde a la expresión

$$E\left[Y\middle|Z\right] = H\left(\beta_0 + \beta_1 z_1 + \beta_2 z_2 + \ldots + \beta_p z_p\right) = \frac{\exp\left(\beta_0 + \beta_1 z_1 + \beta_2 z_2 + \ldots + \beta_p z_p\right)}{1 + \exp\left(\beta_0 + \beta_1 z_1 + \beta_2 z_2 + \ldots + \beta_p z_p\right)} \tag{3.20}$$

Para la estimación del modelo anterior se ha empleado el algoritmo iterativo "*Fisher Scoring*" que exponemos a continuación (MacCullagh and Nelder (1989)).

3.2.2.1 ALGORITMO FISHER SCORING

Para fijar la notación se denota por *Z* la matriz de diseño de variables predictorias dada por

$$\mathbb{Z} = \begin{pmatrix} 1 & Z_{1,1} & \cdots & Z_{1,p} \\ 1 & Z_{2,1} & \cdots & Z_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & Z_{n,1} & \cdots & Z_{n,p} \end{pmatrix} \tag{3.21}$$

y por $Y^T = \left(Y_1, Y_2, \cdots, Y_n\right)$. Los pasos del algoritmo son los que siguen:

Paso 1: Inicializar *k*=0. En base a la muestra $\left\{\left(Z_i, \tilde{Y}_i^k\right)\right\}_{i=1}^n$ con $\tilde{Y}_i^k = H^{-1}\left(Y_i\right)$ se estima el modelo de regresión lineal múltiple obteniéndose el vector de parámetros inicial dado por

$$\hat{\beta}^k = \left(\mathbb{Z}^T\mathbb{Z}\right)^{-1}\mathbb{Z}^T\tilde{Y}^k \tag{3.22}$$

siendo $\tilde{Y}^k = \left(\tilde{Y}_1^k, \tilde{Y}_2^k, \cdots, \tilde{Y}_n^k\right)^T$

Paso 2: Calcular el vector de trabajo dado por (3.23)

$$\tilde{Y}_i^k = \eta_i^k + \left(Y_i - \mu_i^k\right)\frac{\partial \eta_i^k}{\partial \mu_i^k} \tag{3.23}$$

con $\eta_i^k = \hat{\beta}_0^k + \sum_{j=1}^{p}\hat{\beta}_j^k Z_{i,j}$ y $\mu_i^k = H\left(\eta_i^k\right)$

A partir de la muestra, con vector de pesos asociados (3.24), se calcula el modelo de regresión lineal ponderado de modo que se obtiene la actualización (3.25), siendo $W^k = diag\left(W_1^k, W_2^k, \cdots, W_n^k\right)$

$$W_i^k = \left(\partial\mu_i^k / \partial\eta_i^k\right)^2 \cdot \left[Var\left(Y_i \middle| \mu_i^k\right)\right]^{-1} \tag{3.24}$$

$$\hat{\beta}^{k+1} = \left(\mathbb{Z}^T W^k \mathbb{Z}\right)^{-1}\mathbb{Z}^T W^k \tilde{Y}^k \tag{3.25}$$

Paso 3: Se repite el paso 2 con k=1,2,3,… hasta que se alcanza el orden de convergencia dado por

$$\frac{\left\|\hat{\beta}^{k+1} - \hat{\beta}^k\right\|}{\left\|\hat{\beta}^k\right\|} \le \varepsilon \tag{3.26}$$

para algún ε suficientemente pequeño.

Dado un nuevo punto z_0 que no necesariamente tiene que pertenecer a la muestra original, la ecuación de predicción será la dada por

$$\begin{aligned}\hat{E}\left[Y \middle| Z_0\right] = \mu_0^k &= H\left(\left[\left(\mathbb{Z}^T W^k \mathbb{Z}\right)^{-1}\mathbb{Z}^T W^k \tilde{Y}^k\right]^T \begin{pmatrix}1\\ Z_0\end{pmatrix}\right) \\ &= \frac{\exp\left(\left[\left(\mathbb{Z}^T W^k \mathbb{Z}\right)^{-1}\mathbb{Z}^T W^k \tilde{Y}^k\right]^T \begin{pmatrix}1\\ Z_0\end{pmatrix}\right)}{1+\exp\left(\left[\left(\mathbb{Z}^T W^k \mathbb{Z}\right)^{-1}\mathbb{Z}^T W^k \tilde{Y}^k\right]^T \begin{pmatrix}1\\ Z_0\end{pmatrix}\right)}\end{aligned} \tag{3.27}$$

3.2.3 Escenarios de Simulación

Este estudio de simulación sirve para comparar las estimaciones obtenidas mediante el algoritmo Fisher -Scoring y las dadas por la red neuronal seleccionada, un perceptrón con una capa oculta, con función de activación logística en ambas capas.

El vector regresor es bidimensional, de modo que las variable unidimensionales son independientes, idénticamente distribuidas, con distribución uniforme en el intervalo (-2,2). La variable respuesta Y¸ es tal que la distribución de Y condicionada a los valores de Z siguiese una Bernouilli (3.28), con el fin de estimar la esperanza de dicha variable (3.29).

$$Y \mid \mathbf{Z} = \mathbf{z} \in \text{Bernouilli}\left(H\left[f_1(z_1) + f_2(z_2)\right]\right) \tag{3.28}$$

$$E\left[Y \mid \mathbf{Z}\right] = P\left[Y = 1 \mid \mathbf{Z}\right] \in [0,1] \tag{3.29}$$

Se crearon dos escenarios diferentes, uno más propicio para el modelo lineal general y otro más general, más favorable para la estimación con redes neuronales.

Escenario 1

En este escenario las funciones sobre las variables son lineales, con una función link logística.

$$f_1(z_1) = z_1 \tag{3.30}$$

$$f_2(z_2) = z_2 \tag{3.31}$$

$$H(\eta) = \frac{\exp(\eta)}{1+\exp(\eta)} \tag{3.32}$$

Este escenario es propicio para el modelo GLM puesto que responde a su estructura, mientras que la red neuronal, al ser más un modelo más general puede no obtener tan buenos resultados en este caso.

Escenario 2

En este escenario se ha añadido perturbaciones, de modo que no responde ya a un modelo GLM, lo que puede hacer que este modelo presente una comportamiento inadecuado. Este es el escenario en el que la red deberá mostrar su capacidad de modelización.

$$f_1(z_1) = z_1 + 2\sin\left(\pi\left(2 + \frac{z_1}{2}\right)\right) \tag{3.33}$$

$$f_2(z_2) = z_2 \tag{3.34}$$

$$H(\eta) = \frac{\exp\left[\eta + \sin\left(\pi\left(1 + \frac{\eta}{4}\right)\right)\right]}{1+\exp\left[\eta + \sin\left(\pi\left(1 + \frac{\eta}{4}\right)\right)\right]} \tag{3.35}$$

Este escenario es propicio para el modelo GLM puesto que responde a su estructura, mientras que la red neuronal, al ser más un modelo más general puede no obtener tan buenos resultados en este caso.

Se han generado 1.000 muestras de tamaño 1.000, para el entrenamiento o estimación. La evaluación del comportamiento de los modelos en cada muestra se realizará con 250 puntos

del soporte de las covariables generados de modo independiente a la muestra empleada para la estimación. La medida del error se realiza con la deviance.

3.2.4 Resultados y Discusión

La tabla 3.4 muestra los resultados de la deviance media en los conjuntos de validación, para ambos modelos en los dos escenarios de simulación. Se aprecia que en el escenario favorable al modelo GLM éste mejora los resultados que la red neuronal, pero sólo ligeramente. En el escenario desfavorable para el GLM este presenta un resultados peor que el de la red. En este segundo caso la estructura de probabilidad es más compleja, por lo que ambos modelos de predicción presentan resultados menos exactos.

	DEVIANCE	
MODELO	ESCENARIO 1	ESCENARIO 2
GLM	1.195	1.375
RED	1.199	1.227

Tabla 3.4 Deviance media

Las figuras 3.4 y 3.5 muestran la comparación de los resultados de los 1000 conjuntos de validación, mediante diagramas de cajas, para cada uno de los escenarios. Se aprecia que en el caso del primer escenario la red funciona ligeramente peor, pero la diferencia es claramente no significativa. En el segundo escenario ambos modelos presentan peores resultados, pero claramente la red neuronal es mejor que el modelo GLM.

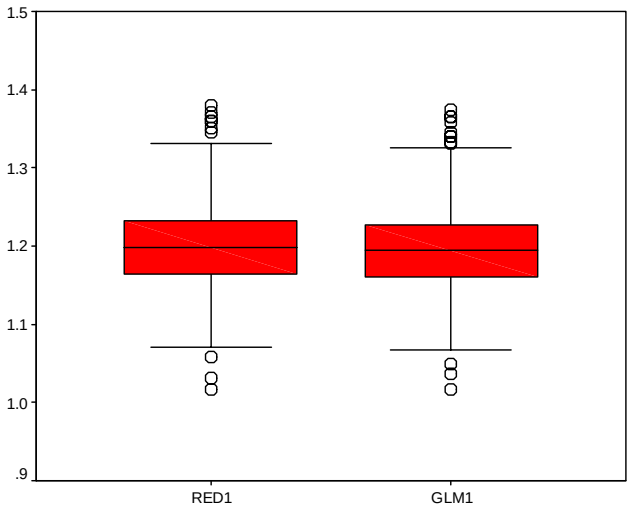

Figura 3.4 Comparación de la Deviance para los dos modelos en el Escenario 1

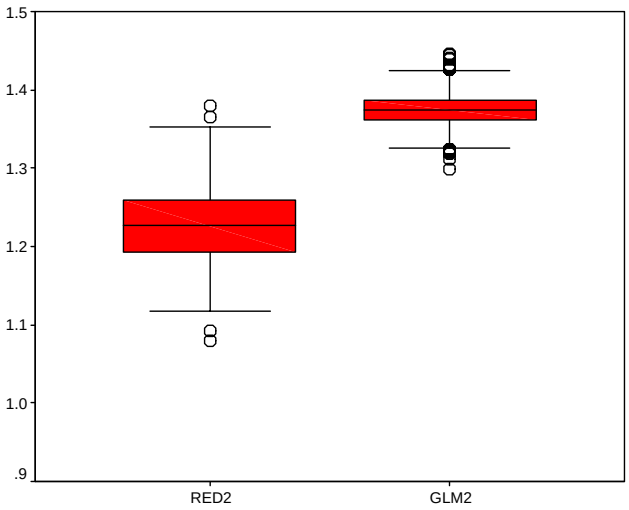

Figura 3.5 Comparación de la Deviance para los dos modelos en el Escenario 2

Para analizar con más detenimiento el resultado de la red neuronal, en las figuras 3.6 y 3.7 se muestran las superficies de probabilidad en ambos escenarios, a fin de comparar la real con la estimada con la red neuronal.

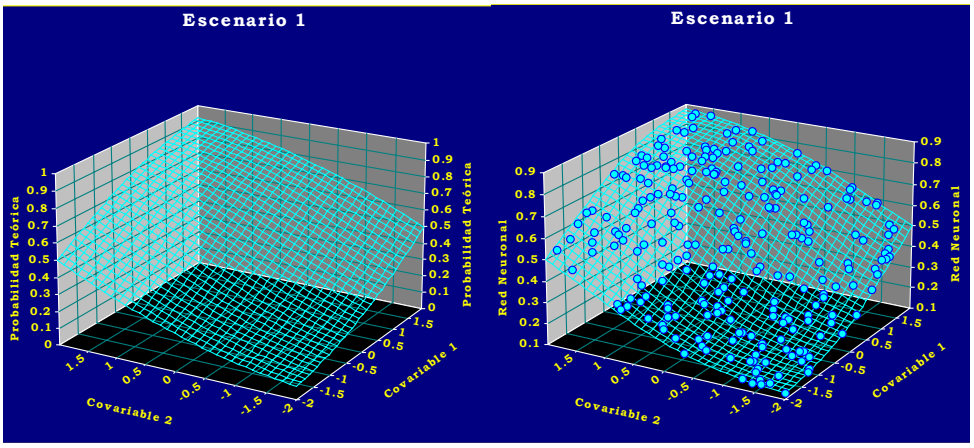

Figura 3.6 Comparación de superficies de probabilidad teórica y de red neuronal en el Escenario1

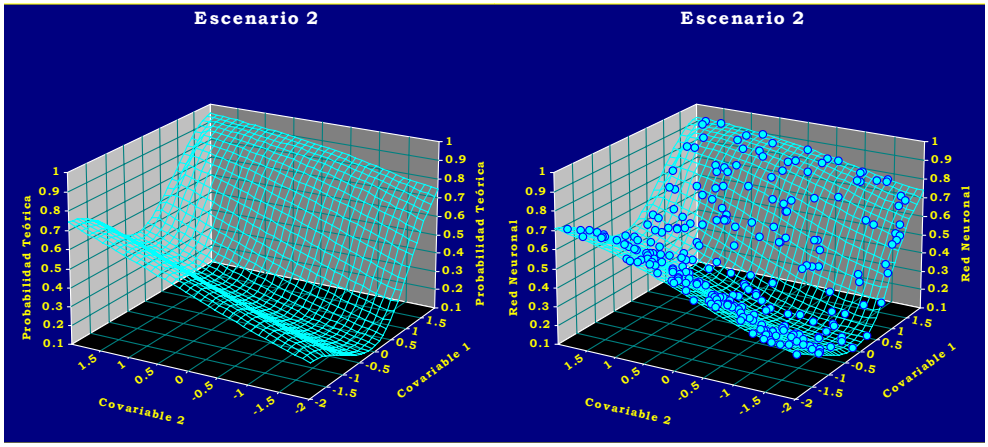

Figura 3.7 Comparación de superficies de probabilidad teórica y de red neuronal en el Escenario 2

Las redes neuronales han demostrado ser competitivas frente a otros métodos de estimación de variables binarias, tanto en escenarios favorables a estos métodos como en escenarios más generales.

3.3 Conclusiones

En este capítulo se ha estudiado el comportamiento de las redes neuronales ante problemas de clasificación. Se han presentado dos ejemplos, uno con datos reales y otro con datos simulados de modelos respondiendo a dos modelos distintos.

Se han obtenido resultados satisfactorios en ambos entornos, mostrando las capacidad de clasificación de las redes neuronales; destaca el comportamiento en el caso de simulación donde en escenarios favorables a otros modelos de predicción han mostrado un mejor comportamiento.

3.4 Bibliografía

Aira, M.J., Jato, V., Iglesias, I. (1998) *Alnus* and *Betula* pollen content in the atmosphere of Santiago de Compostela. North-Western Spain (1993-1995). Aerobiologia V.14(2,3), pp. 135-140.

Aira, M.J., Ferreiro, M., Iglesias, I., Jato, V., Marcos, C., Varela, S., Vidal, C. (2001) Aeropalinología de cuatro ciudades de Galicia y su incidencia sobre la sintomatología alérgica estacional. Actas XIII Simposio de la A.P.L.E..Cartagena.

Agresti, A. (1990) *Categorical Data Analysis*. Wiley.

Andersen, T.B. (1991) A model to predict the beginning of the pollen season. Grana V.30, pp. 269-275.

Arenas, L., González, C., Tabarés, J.M., Iglésias, I., Méndez, J., Jato, V. (1996) Sensibilización cutánea a pólenes en pacientes afectos de rinoconjuntivitis-asma en la población de Ourense en el año 1994-95. 1st. European Symp. On Aerobiol. Santiago de Compostela 93-94.

Atkinson, H., Larsson. K,A, (1990) A 10 year record of the arboreal pollen in Stockholm, Sweden. Grana V.29, pp. 229-237.

Buja, A., Hastie, T.J. and Tibshirani, R.J. (1989) Linear Smoothers and Additive Models. Annals of Statistics, V.17, pp. 453-555.

Bishop, C. (1995) Neural Networks for Pattern Recognition, Oxford:Clarendon Press.

Caramiello, R., Siniscalco, C., Mercalli, L., Potenza, A. (1994) The relationship between airborne pollen grains and unusual weather conditions in Turin (Italy) in 1989, 1990 and 1991. Grana, V.33, pp. 327-332.

Castellano-Méndez, M., Aira, M.J., Iglesias, I., Jato, V., González-Manteiga, W. (2005). Artificial Neural Network As Useful Tool To Predict The Risk Level Of The Betula Pollen In The Air. International Journal of Biometeorology, V.49(5), pp.310-316.

Chakraborty, K., Mehrotra, K. (1992) Forecasting the behaviour of a multivariate time series using neural networks. Neural Networks, V.5, pp. 961-970

Chauvin, Y., Rumelhart, D.E. (1995) Backpropagation: Theory, Architectures, and Applications. Lawrence Erlbaum Associates, Inc.

Clot, B. (2001) Airborne birch pollen in Neuchâtel (Switzerland): onset, peak and daily patterns. Aerobiologia, 17(1), pp. 25-29.

Corsico, R. (1993) L'asthme allergique en Europe. In FTM Spieksma, N Nolard, G Frenguelli D Van Moerbeke (eds), (1993). Pollens de l'air en *Europe*. UCB, Braine-l'Alleud, pp. 19-29.

Costa, M., Higueras, J., Morla, C. (1990) Abedulares de la Sierra de San Mamed (Orense, España). Acta Bot. Malacitana, v.15, pp. 253-265.

Cybenko, G. (1989) Approximations by Superpositions of a Sigmoidal Function. Math. Contr. Signals, Systems Vl2, pp. 303-314

D'Amato,G., Spieksma, F.Th.M. (1992) European allergenic pollen types. Aerobiologia, V.8, pp. 447-450.

Domínguez, E. (1995) La Red Española de Aerobiología. Monografía REA, V.1, pp. 1-8.

Dopazo, A. (2001) Variación estacional y modelos predictivos de polen y esporas aeroalergénicos en Santiago de Compostela. Tesis Doctoral. University of Santiago de Compostela.

Ekebom, A., Verterberg, O., Hjelmroos, M. (1996) Detection and quantification of airborne birch pollen allergens on PVDF membranes immunoblotting and chemiluminescence. Grana, V.35, pp. 113-118.

Goldberger, A.S. (1973) Correlations between *binary choices and probabilistic predictions.* Journal of the American Statistical Association, 68:84

Härley, W. (1991) Smoothing Techniques with implementation in S. Springer-Verlag.

Hastie, T. (1987) A closer look at the deviance. The American Statistician, V.41(1), pp. 16-20.

Hastie, T.J. and Tibshirani, R.J. (1990) Generalized Additive Models. Chapman and Hall.

Hidalgo, P.J., Mangin, A., Galán, C., Hembise, O., Vázquez, L.M., Sánchez, O. (2002) An automated system for surveying and forecasting Olea pollen dispersion. Aerobiologia, V. 18, pp.23-31

Hilera González, J.R. y Martines Hernando, V.J. (1995) Redes Neuronales Artificiales. Fundamentos, Modelos y Aplicaiones. Ra-ma

Hjelmroos, M. (1991) Evidence of long-distance transport of Betula pollen. Grana, V.30, pp. 215-228.

Hornik, K.M., Stinchcombe, M., White, H. (1989) Multilayer feedforward networks are universal approximators. Neural Networks, V.2, pp. 359 – 366

Izco, J. (1994) O bosque Atlántico. In Vales C. (ed.). Os Bosques Atlánticos Europeos. Bahía edic. La Coruña, pp. 13-49

Jato, V., Aira, M.J., Iglésias, M.I., Alcázar, P., Cervigón, P., Fernández, D., Recio, M., Ruíz, L., Sbai, L. (2000) Aeropalynology of birch (*Betula* sp.) in Spain. Polen, V.10, pp. 39-49.

Laaidi, M. (2001) Regional variations in the pollen season of Betula in Burgundy: two models for predicting the start of the pollination. Aerobiologia, V.17(3), pp. 247-254.

Larsson, K. (1993) Prediction of the pollen season with a cumulated activity method. Grana, V.32, pp. 111-114.

Latalowa, M., Mietus, M., Uruska, A. (2002) Seasonal variations in the atmospheric *Betula* pollen count in Gdansk (southern Baltic coast) in relation to meteorological parameters. Aerobiologia, V.18, pp. 33-43.

Lewis, W.H., Vinay, P., Zenger, V.E. (1983) Airborne and allergenic pollen of North America. The Jones Hopkins Univ. Press.

Looney, C.G. (199 7).Pattern Recognition and neural networks. Cambridge University Press.

McCullagh, P., Nelder, J.A. (1989) General Linear Models, Second Edition, London: Chapman & Hall

Méndez, J. (2000) Modelos de comportamiento estacional e intradiurno de los pólenes y esporas de la ciudad de Ourense y su relación con los parámetros meteorológicos. Tesis Doctoral. Universidad de Vigo.

Moreno, G. (1990) In Castroviejo S. Edit. Flora Ibérica. Vol. II. Real Jardín Botánico. C.S.I.C. Madrid.

Moore, P.D., Webb, J.A. (1978) An illustrated guide to pollen analysis. Hodder & Soughton.

Moseholm, L., Weeke, E.R., Petersen, B.N. (1987) Forecast of pollen concentration of Poaceae (Grasses) in the air by Time Series Analysis. Pollen Spores, V.2(3), pp. 305-322

Negrini, A.C., Voltolini, S., Troise, C., Arobba, D. (1992) Comparison between Urticaceae (*Parietaria*) pollen count and hay fever symptoms: assessment of a threshold value. Aerobiologia, V. 8, pp. 325-329

Norris-Hill, J., Emberlin, J. (1991) Diurnal variation of pollen concentration in the air of north-central London. Grana, V.30, pp. 229-234.

Opsomer, J. (2000) Asymptotic Properties of Backfitting Estimators. Journal of Multivariate Analysis, V.73, pp. 166-179.

Opsomer, J.D. y Kauermann, G. (2000) Weighted Local Polynomial Regression, Weighted additive models and local scoring. Preprint #00-7, Department of Statistics, Iowa State University

Opsomer, J.D. and Ruppert, D (1997) Fitting a bivariate additive model by local polynomial regression. Annals of Statistics, V.25, pp. 186-211

Park, J., Sandberg, I.W. (1991) Universal approximation using radial basis function networks. Neural Computation, V.3, pp. 246-257.

Rantio-Lehtimäki, A., Pehkonen, E., Yli Panula, E. (1996) Pollen allergic symptoms in the off season?. In Aira, M.J., Jato, V., Iglesias, I., Calán, C. Edit. Compostela Aerobiology V.96, pp. 91-92.

Ranzi, A., Lauriola, P., Marletto, V., Zinozi, F. (2000) Forecasting Airborne Pollen Concentrations: Development of Local Models. Abstracts of Second European Symposium on Aerobiology, 43.

Ripley, B.D. (1996) Pattern Recognition using Neural Networks. Cambridge University Press.

Rodríguez-Rajo, F.J. (2000) El polen como fuente de contaminación ambiental. Tesis Doctoral. Universidad de Vigo.

Rosenblatt, F. (1958) The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, V.65, pp. 386-408.

Ruffaldi, P., Greffier, F. (1991) Birch (*Betula*) pollen incidence in France (1987-1990). Grana, V.30, pp. 248-254.

Rumelhart, D.E., Hinton, G.E., Williams, R.J. (1986) Learning internal representations by error propagation, in Rumelhart DE McClelland JL eds. (1986), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, V.1, pp. 318-362, Cambridge, MA: The MIT Press.

Ruppert, D., and Wand, M.P. (1994) Multivariate Locally weighted least minimum squares regression. Annals of Statistics, V.22, pp. 1346-1370

Sánchez-Mesa, J.A., Galán, C., Martínez-Heras, J.A., Hervás-Martínez, C. (2002) The use of a neural network to forecast daily grass pollen conentration in a Mediterranean region: the southern part of the Iberian Peninsula. Clin. Exp. Allergy, 32

Sarle, W.S. (1994) Neural network and statistical models. In proceedings of the 19th annual SAS Users Group International Conference, Cray LC.

Spieksma, F.Th.M., Frenguelli, G., Nikkels, A.H., Mincigrucci, G., Smithius, L.O.M.J., Bricchi, E., Dankaart, W., Romano, B. (1989) Comparative study of airborne pollen concentrations in central Italy and The Netherlands (1982-1985). Grana, V.28, pp. 25-36.

Spieksma, F.Th.M. (1990) Pollinosis in Europe: new observations and developments. Rev Paleobot. Palynol, V.64, pp. 35-40.

Spieksma, F.Th.M., Emberlin, J.C., Hjelmroos, M., Jäger, S., Leuschner, R.M. (1995) Atmospheric birch (*Betula*) pollen in Europe: Trends and fluctuations in annual quantities and the starting dates of the seasons. Grana, V.34, pp. 51-57.

Wallin, J.E., Segerström, V., Rosenhall, L., Bergmann, E., Hjelmroos, M. (1991) Allergic symptoms caused by long distance transported birch pollen. Grana, V.30, pp. 256-268.

Wand, M.P. and Jones, M.C. (1995). Kernel Smoothing. Chapman and Hall

Wihl, J.A., Ipsen, B., Nüchel, P.B., Munch, E.P., Janniche, E.P., Lövenstein, H. (1998) Immunotherapy with partially purified and standardized tree pollen axtracts. Allergy, V.43, pp. 363-369.

Capítulo 4. Aplicación de Redes Neuronales a Problemas de Control

RESUMEN

Los procesos de producción y reciclado son hoy en día base fundamental de un gran número de empresas. El control de estos procesos constituye hoy en día un requisito imprescindible no sólo en el día a día de la industria sino también está más presente en la vida cotidiana. Los sistemas de control son necesarios para mantener estable el funcionamiento de cualquier proceso complejo. En este capítulo se introducirán los conceptos fundamentales del control de procesos, tanto desde el punto de vista tradicional asociado a la ingeniería como desde el punto de vista estadístico. Asimismo se plantearán diferentes posibilidades para la integración de las posibilidades de las redes neuronales dentro de distintos sistemas de control. Se ilustrarán las posibilidades más importantes con aplicaciones al control de procesos reales, en el entorno del tratamiento anaeróbico de aguas, y de un proceso siderúrgico.

Parte de los resultados que se detallan en este capítulo están recogidos en Ruiz *et al.*, 2005ª, Ruiz, *et al.*, 2005b, Castellano *et al.*, 2007 y Molina *et al.*, 2009.

4.1 Introducción al Problema de Control

Los sistemas de control constituyen una herramienta fundamental en el funcionamiento de multitud de procesos en ámbitos de trabajo muy diferentes. Entre las ventajas generales asociadas a estos sistemas destaca poder garantizar la bondad y homogeneidad del producto, así como la disminución de costes, bien por la optimización del proceso, bien por la disminución de pérdidas asociadas a productos defectuosos u otros problemas que un sistema de control y monitorización detecta de forma temprana.

4.1.1 Nociones básicas de control

A lo largo del capítulo, a la hora de desarrollar cualquier problema en el ámbito del control, van a aparecen de modo recurrente ciertos conceptos e ideas que es necesario definir adecuadamente.

Un sistema de control se encuentra en *lazo abierto* o en *manual* cuando el controlador no se encuentra conectado al proceso, de modo que sus acciones no se convierten en cambios en el proceso; mantener un sistema en lazo abierto puede ser útil o necesario en determinadas ocasiones, por ejemplo para evaluar las propuestas de un controlador en fase de desarrollo, o en situaciones extremadamente delicadas de control, cuando es necesario u seguimiento pormenorizado, que a menudo requiere que un operador revise las propuestas del controlador antes de que se efectúen. Muchos procesos suelen funcionar en abierto o manual durante su

puesta en marcha o su apagado. Un sistema se encuentra en *lazo cerrado* o *automático* cuando efectivamente es el controlador el que determina los cambios que se producen en el sistema.

La variable *del proceso a controlar*, o *controlada*, es aquella que se desea mantener en un valor o rango deseados. En realidad si se desea ser estricto esta variable nunca se conoce, sólo se tiene la variable medida con un instrumento.

El *punto de consigna*, denominado también *referencia*, es el valor deseado para la variable a controlar. Estos puntos pueden ser constantes variar poco (control - regulador) o pueden ser muy variables (control – servomecanismo).

La variable *de control* o *manipulable* es aquella que se puede manejar a voluntad y es empleada para mantener la variable a controlar en el punto de control; es la variable que se emplea para compensar las perturbaciones que sufre el proceso. Al igual que ocurría con la variable controlada, no se manipula la variable real, sino su transformada en señal medida.

Finalmente las variables de *perturbación* son aquellas que afectan a las variables controladas pero son externas al sistema, de modo que no es posible actuar sobre ellas. Algunas pueden ser cuantificadas pero otras no; algunos autores consideran los cambios de consigna como variables de perturbación.

El *controlador* es el sistema que a partir de los valores de la variable a controlar, la variable manipulable y la consigna calcula la *acción de control* de acuerdo con algún algoritmo de control (en la subsección siguiente se detallarán los tipos de controladores más usuales). Esta acción de control se traduce en un valor de la variable manipulable, que se convertirá en un acto físico en el elemento final de control o actuador.

4.1.2 Tipo de modelos de control

Existen diferentes modos de clasificar los distintos algoritmos o técnicas de control. Una posible clasificación distingue entre técnicas de *control clásico* y técnicas de *control avanzado*. Las de control clásico suelen estar enfocados a lazos sencillos, que en numerosas ocasiones tendrán que tener controladores superiores. Otro tipo de clasificación se centra en la separación según el *nivel* de control que efectúan sobre el proceso. La clasificación del proceso de control se divide en niveles según la complejidad del nivel de automatización adquirido, en *control regulatorio básico, control regulatorio avanzado, control multivariante* y *control con optimización en línea*. Subir el nivel de automatización requiere más complejidad y costos económicos, pero también permite acercar el sistema a su funcionamiento óptimo. A continuación se amplían estas clasificaciones con más detalle.

4.1.2.1 Control Clásico frente a Control Avanzado

4.1.2.1.1 Control Clásico

Existen cuatro tipos fundamentales de controladores que se encuadran dentro del control clásico: *control por retroalimentación*, el *control anticipado* o por acción precalculada, el *control en cascada* y el *control adaptativo*. El nexo de unión entre estos métodos de control es la necesidad de disponer de un modelo del proceso, que puede ser más o menos sofisticado, según los objetivo requeridos y el tipo de controlador elegido (Olsson y Newell, 1999).

Control por Retroalimentación

En el control por retroalimentación la actuación del controlador se calcula a partir de la diferencia existente entre la variable a controlar y la consigna. En el esquema de la figura 4.1 se muestra el funcionamiento de un lazo de control por retroalimentación. En este esquema formal se incluyen elementos como el sensor que mide la variable a controlar, el actuador que traducirá el cálculo del controlador en una modificación sobre la variable manipulable y las perturbaciones que influyen sobre el proceso. Se detalla la lectura del esquema en este primer caso sencillo para facilitar su comprensión de este tipo de esquemas. El símbolo circular representa una operación de modo que en este caso, y en concordancia con los signos indicados se calcula la diferencia entre la consigna y el valor de la variable a controlar medio por el sensor; a partir de está única información el controlador calcula la acción correctora pertinente, el actuador la ejecuta, y el proceso sigue su curso, dando lugar a la variable controlada, entre otras. Esta variable de nuevo será evaluada con el sensor, y comparada con la consigna para permitir al controlador calcular la nueva acción de control.

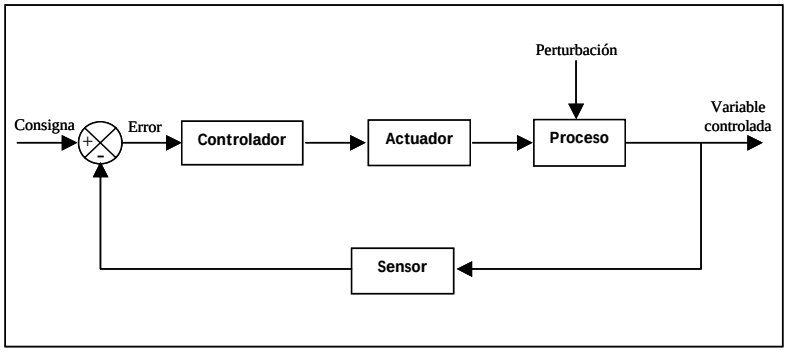

Figura 4.1 Esquema de Control por Retroalimentación

Este tipo de control resulta esquemática e ideológicamente sencillo, y permite compensar las perturbaciones de modo que la variable controlada se acerque a su objetivo, la consigna. Su principal inconveniente es que actúa a posteriori, una vez que las perturbaciones del sistema han alejado la variable a controlar de la consigna, por lo que en la práctica no evita las desviaciones, sino que las reduce.

Cómo funciona el actuador es también un factor importante en cualquier tipo de control. El controlador puede actuar de modo binario, encendiendo o apagando el actuador (control on/off), con el fin de mantener la variable a controlar dentro de un intervalo; en este caso el actuador enciende una válvula al llegar a cierto valor y la apaga al llegar a otro. En este tipo de actuadores la variable a controlar no alcanza un valor consigna sino que se mantiene en un intervalo, en principio, generado entorno a la consigna. Esta variabilidad alrededor del valor de consigna puede ajustarse modificando los valores umbrales que determinan el encendido y el apagado. Hay procesos y variables que responden bien a este tipo de actuadores, pero en otros casos no es suficiente, bien porque la estabilidad del sistema requiere un manejo más fino, bien porque la variabilidad del sistema es tal que la excesiva sucesión de cambios puede originar problemas en el equipo.

En estos casos es necesario recurrir a métodos de control más elaborados. Uno de los problemas del controlador on/off es que la actuación no es proporcional al valor de la variable manipulada, ni por lo tanto al error cometido respecto a la consigna. Una primera mejora es que la actuación sea proporcional al error cometido e(t), (control proporcional, P,) a través de una constante de proporcionalidad Kc, también llamada ganancia del controlador. Como la modificación es proporcional al error, lo que ocurre es que la variable se va acercando a la consigna sin llegar a alcanzarla. Esta diferencia que no se logra solventar se denomina off-set, y está determinada por el valor de Kc, de modo que es necesario seleccionar (sintonizar) Kc para obtener la mejor respuesta posible ante las diferentes perturbaciones.

La siguiente mejora consiste en eliminar el off-set. Para ello se añade a la acción de control calculada por el controlador un término asociado al error acumulado durante un intervalo de tiempo, término integral, medido a través de la integral del error. De este modo surge el control proporcional integral , PI. Este nuevo control no adolece de la existencia del término off-set, pero el nuevo término genera oscilaciones en el control, cuya importancia depende de la constante asociada al término integral. Para compensar este efecto, y disminuir las oscilaciones se recurre a la inclusión de un nuevo término, en este caso asociado a la velocidad de cambio del error (término derivativo), surgiendo así el *control proporcional integral derivativo*, *PID*. Es posible combinar de diferentes maneras la aparición de estos términos, aunque los controladores más comunes son los mencionados anteriormente.

Algunos términos no han de aparecer solos por distintas razones. En el caso del integral las oscilaciones que genera son importantes, por lo que es necesario emplearlo combinado, bien un controlador PI o bien un PID. Cuando el sistema alcanza una situación constante el término derivativo es nulo, por lo que no hay respuesta de control. Este término solo no trabaja bien en estas circunstancias, por lo que es necesario combinarlo con alguno de los otros términos dando lugar a un controlador PD o PID. Finalmente cuando la variable a controlar presenta mucha variabilidad o ruido, se suelen obtener valores altos en el término derivativo, con las consiguientes oscilaciones; es por ello que bajo estas condiciones no se emplea el término derivativo.

Si se denomina m a la variable manipulada, los valores de esta variable calculados por un controlador PID responden a (4.1). Las constantes que aparecen en el modelo son las constantes de proporcionalidad de los diferentes términos, *Kc* la del término proporcional, como ya se había señalado, τ_I la del término integral y τ_D la del derivativo.

$$m = K_c \cdot \left(e + \frac{1}{\tau_I} \cdot \int_0^t e \cdot d\tau + \tau_D \cdot \frac{de}{dt} \right) \tag{4.1}$$

La elección del controlador depende en gran medida del tipo de problema al que se va a aplicar, así como de las variables controlada y manipulable seleccionadas.

Para obtener un buen funcionamiento del controlador seleccionado es necesaria una adecuada selección de los parámetros del mismo, proceso que se denomina *sintonizar* el controlador. Los métodos más comunes para llevar a cabo esta tarea fueron propuestos por

Ziegler y Nichols (1942). Estos métodos son el *método de la curva de reacción del proceso* y el *método de la ganancia última*. El primero de ellos se aplica en un sistema en lazo abierto, y el segundo requiere un experimento en lazo cerrado. Cada uno presenta ciertas ventajas e inconvenientes que han de ser evaluadas según las características del proceso.

En la práctica no se tiene la curva continua, e(t), sino sólo los valores en ciertos momentos. Por la expresión continua (4.1) puede sustituirse, basándose en las aproximaciones de (4.2), por su variante discreta, (4.3).

$$\int_0^t e \cdot d\tau \approx \sum_{k=1}^{n} e_k \cdot \Delta t$$

$$\frac{de}{dt} \approx \frac{e_n - e_{n-1}}{\Delta t} \tag{4.2}$$

$$m = K_c \cdot \left(e_n + \frac{\Delta t}{\tau_I} \cdot \sum_{k=1}^{n} e_k + \frac{\tau_D}{\Delta t} \cdot \left(e_n - e_{n-1} \right) \right) \tag{4.3}$$

Control Anticipativo o por Acción Precalculada

La idea de realizar correcciones a partir del error observado tiene el inconveniente de que es necesario esperar a que se produzca el error para efectuar las correcciones. Una idea interesante se basa en la posibilidad de adelantarse a este error y actuar sobre el proceso en función de las perturbaciones observadas, antes de que estas se propaguen por el sistema. El funcionamiento de esta técnica de control consiste en considerar las variables de entrada del proceso y a partir de ellas calcular el valor que ha de tomar la variable manipulada para que la variable controlada esté cercana al valor de consigna. Un esquema de su funcionamiento puede verse en la figura 4.2. Esta técnica tiene la ventaja de que al menos teóricamente es posible alcanzar un control perfecto del sistema, si bien en la práctica esto no es posible, pues sería necesario conocer todas las posibles perturbaciones del sistema, algunas de las cuales pueden no ser ni siquiera medibles, y actuar respecto a todas ellas.

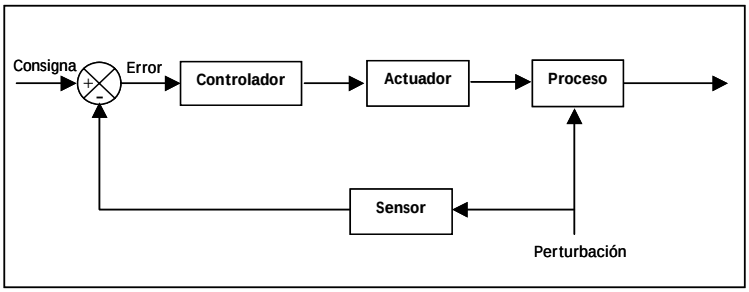

Figura 4.2 Esquema de Control Anticipativo

Este método tiene como principal inconveniente la necesidad de disponer de un modelo que sea capaz de establecer la relación entre las perturbaciones y la variable a controlar, y esto no siempre es posible. Es por esto que en ocasiones se emplean conjuntamente las técnicas de control anticipativo y por retroalimentación con el fin de que en control de retroalimentación permita conocer y mantener la variable controlada en el valor nominal,

y eliminar cualquier problema surgido de la determinación de la relación entre las perturbaciones y la variable a controlar, así como compensar el efecto de las perturbaciones no medidas (Lim y Lee, 1991). La figura 4.3 muestra un ejemplo de control mixto (Smith y Corripio, 1991).

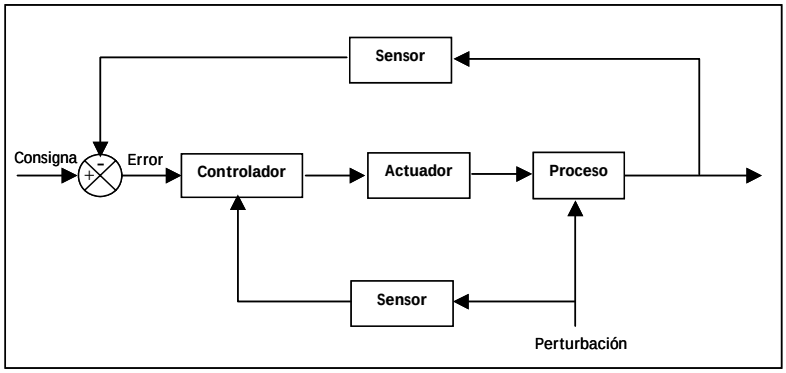

Figura 4.3 Esquema de un lazo de control mixto de retroalimentación y de control anticipado

Estos controles se refieren principalmente a una variable unidimensional. Si la variable a controlar es multidimensional es necesario establecer varios lazos de control, en lo que pasa a ser un nuevo tipo de control.

Control de lazos múltiples

El análisis anterior puede extenderse a varias variables controladas, bien de forma paralela o en cascada. Si se desea que sean lazos en paralelo con diferentes variables manipulables es necesario que las variables sean independientes entre sí. Si las variables se encuentran relacionadas no se deben establecer lazos simples en paralelo, debido a la dificultad a la hora de determinar la respuesta conjunta frente a cambios en las variables manipuladas. En estos casos resulta más sencillo emplear un control en cascada, con un lazo de control interno y otro externo que ajusta la consigna del lazo interno.

Control Adaptativo

Los sistemas de control se aplican a procesos cuyo funcionamiento y dinámica son susceptibles de sufrir variaciones importantes a lo largo del tiempo. Esta es una de las causas de que en ocasiones sea necesario resintonizar los parámetros del controlador y los valores de consigna al nuevo estado del sistema. Los métodos de control adaptativos constituyen una metodología de control que intenta resolver de modo automático estos problemas, a través de algoritmos de estimación de parámetros.

Es posible distinguir entre diferentes tipos de sistemas adaptativos según la técnica empleada para sintonizar los parámetros. Por una parte están los *controladores autosintonizables* (STR), en los que los parámetros del controlador se calculan a través del estudio de las respuestas del proceso a pequeñas perturbaciones generadas por el sistema. Por otra están los *controladores con modelo de referencia* (MRAS), que ajustan sus parámetros a partir de las diferencias entre el comportamiento real del proceso y el

comportamiento esperado, calculado a partir de un modelo del sistema. Este modelo presenta dos lazos de control, uno externo de adaptación, que ajusta los parámetros del modelo (unívocamente relacionados con los del controlador) para minimizar las diferencias entre los valores reales y predichos por el modelo, y el interno, que constituye un modelo de control por retroalimentación.

4.1.2.1.2 Control Avanzado

Los sistemas de control avanzado contienen un control de bajo nivel, al que se le superpone este control de alto nivel, que realiza tareas de supervisión del funcionamiento del sistema. Entre las tareas asignadas a un controlador avanzado destacan la monitorización del proceso, control de fallos, validación de las medidas de los sensores, diagnosis del proceso, optimización del proceso, recuperación frente a fallos... (Aynsley ***et al.***, 1993).

Estas técnicas han de basarse en un conocimiento profundo del proceso, que bien puede ser proporcionado por la experiencia de un experto en el proceso, o bien un histórico de datos de entrada / salida representativo del proceso, o en modelos de caja negra. No es tanto la necesidad de tener un modelo detallado del proceso, que puede ser muy complejo, no lineal, etc., como de tener un conocimiento del mismo quizás más general, pero basado en el propio proceso.

La configuración de sistemas de control basados en el conocimiento puede ser divididas en dos grandes grupos, el control experto directo y el control supervisor o indirecto (Konstantinov ***et al.***, 1992). En el directo el módulo basado en el conocimiento está dentro del lazo de control, al nivel de un PID. Muchos de estos controladores están configurados basándose en técnicas de lógica difusa (Reyero y Nicolás, 1995). Los modelos de control supervisor por su parte presentan dos niveles, que tienen asignadas diferentes tareas dentro del control. El sistema no actúa directamente en el control, no proporciona en ningún caso la acción de control deseado, sino que se dedica a supervisar que el sistema de control de bajo nivel funcione correctamente, detecta problemas, cambia consignas,...

Dentro de la metodologías más usuales en el control avanzado están creación de modelos del proceso mediante modelos empíricos funcionales, redes neuronales, algoritmos genéticos, lógica difusa, sistemas expertos basados en el conocimiento, métodos estadísticos o alguna combinación de los anteriores.

4.1.2.2 Según el nivel de automatización

4.1.2.2.1 Control Regulatorio Básico

El control Regulatorio Básico (CRB) constituye el nivel más básico de automatización del proceso y consiste en la implementación de lazos simples de retroalimentación. Presenta los mismos problemas de los que adolece el control de retroalimentación, y por esos mismos motivos es necesario establecer estructuras más sofisticadas que presenten nuevas soluciones frente a los inconvenientes presentados, como la necesidad de trabajar con valores de consigna que ofrezcan un amplio margen de seguridad.

4.1.2.2.2 Control Regulatorio Avanzado

Este segundo tipo de control, el control regulatorio avanzado, CRA, tiene como objetivo principal la mejora de las consignas de modo que lleve al sistema al óptimo de control, en términos económicos, al tiempo que mantiene la seguridad y calidad del proceso.

El CRA emplea técnicas de control que amplían y complementan el control por retroalimentación. En este tipo de control aparecen los lazos en cascada, lazos anticipativos, así como las técnicas de compensación de tiempos muertos y las de control con restricciones.

4.1.2.2.3 Control Multivariante o Multivariable

Algunos autores consideran esta técnica dentro del control avanzado, mientras que otros lo consideran un nivel superior de automatización. La idea ya fue explicada anteriormente al hablar de control de lazos múltiples. Esta técnica de control se construye partiendo de la relación existente entre algunas variables manipulables y las que se desean controlar. Ante la existencia de relaciones múltiples entre ellas resulta necesario la determinación de las relaciones cruzadas a partir de un modelo matricial del proceso, donde se obtienen las variables controladas de salida a partir de las variables manipulables. Esta matriz se invierte para obtener las variables de entrada o manipulables a partir de las controladas. La base radica pues en la estadística multivariante.

4.1.2.2.4 Optimización en Línea

La optimización de un proceso consiste en determinar las condiciones en las el sistema se comporta mejor según algún criterio, en general el máximo beneficio económico. Las condiciones óptimas de aunque éste no siempre es el criterio seleccionado, y a menudo va asociado a otros criterios, en general de calidad. En la mayoría de los casos las condiciones óptimas de operación dependen que variaciones de determinadas variables externas (calidad de las materias, oscilaciones de precios, demandas, estados del sistema,...). Sobre la base de un modelo y una función de beneficio se calculan loa nuevas consignas y las acciones de control adecuadas para alcanzar estas consignas. La periodicidad con que se ajustan las consignas depende del proceso que se esté monitorizando y de las características de las variables empleadas para calcular la función de beneficio.

4.1.3 Diseño del sistema de control

El diseño de un sistema de control constituye un proceso complejo que ha de ser abordado de modo temprano, desde el propio diseño del proceso que se desea controlar, con el fin de garantizar que el proceso sea capaz de responder a los cambios de las variables modificadas, reducir la magnitud y frecuencia de las perturbaciones,... El diseño de control se suele dividir en las siguientes etapas.

- *Definir los Objetivos de Control*

  Estos objetivos pueden estar relacionados con la estabilidad del sistema, la calidad de la producción, los beneficios globales,...

- *Identificar las variables que pueden ser medidas y las que pueden ser manipuladas*

  Es necesario obtener información sobre las variables relacionadas con los objetivos de control, pero no siempre existe la posibilidad de medir esas variables, bien por la naturaleza de la mismas, bien por el coste. En estos casos se buscarán inferir las variables a partir de otras medibles; además es importante la medición de variables de entrada (perturbaciones) y intermedias para desarrollar el sistema de control. La elección de las variables manipulables, que han de estar fuertemente relacionadas con las de control, resulta también vital a la hora de desarrollar un sistema de control adecuado.

- *Seleccionar la Configuración del Sistema de Control*

  Tras identificar los objetivos y las variables de control y manipulables el siguiente paso consiste en definir la técnica de control que se va a emplear. La estructura, las variables involucradas en cada lazo, emparejamiento entre variables manipulables y controladas, etc.

- *Especificar la instrumentación de de monitorización y control*

  Para concretar e implementar la configuración de control es necesario establecer los equipos de medida, los controladores y los actuadores o elementos finales de control.

- *Diseño de los controladores*

  Todos los controladores que se han considerado en la configuración del control han de ser sintonizados y preparados para su correcto funcionamiento. Es necesario pues realizar un estudio del comportamiento dinámico del sistema, de la sensibilidad de las variables controladas respecto a las manipuladas, del tiempo de respuesta,...

En todos estos aspectos la estadística puede y debe jugar un papel fundamental en la toma de decisiones necesarias en cada paso del diseño del sistema de control. La función de beneficios puede ser establecida empleando argumentos estadísticos relacionados con el control de calidad, estudios de investigación operativa, etc. La identificación de las variables medibles relacionadas con las involucradas en la función de beneficios, así como las variables manipulables puede hacerse de modo más sencillo a través de técnicas de regresión, que determinen las dependencias, y técnicas de análisis multivariante de datos que indiquen qué variables son más relevantes en términos de información del sistema. El diseño del sistema de control también puede aprovecharse de la información estadística disponible, por ejemplo en la selección de las variables involucradas en el control anticipativo, el emparejamiento de las variables de control y manipulables, etc. Así mismo el diseño del controlador puede realizarse desde una perspectiva estadística, empleando modelos de regresión, estudios de sensibilidad,...

Al abordar los ejemplos prácticos desarrollados, el estudio se centrará en las aplicaciones realizadas con redes neuronales, pero cabe destacar que el aporte estadístico no se ha reducido sólo a este punto, sino que se ha realizado un estudio más completo, enfocado a la selección de variables, estudios de relaciones, etc.

4.2.Aportaciones de las Redes Neuronales al Problema de Control

Un estudio detallado de los distintos modelos de control permiten estudiar en qué segmentos se puede emplear de modo efectivo la potencia predictora de las redes neuronales. Estudios previos (Narendra *et al.*, 1990; Chen *et al.*, 2000; Wilcox *et al.*, 1995; Guwy, 1997) han mostrado cómo las redes neuronales pueden ser empleadas para abordar los algunos de los principales problemas del control de procesos, como son la incertidumbre, la no linealidad y la complejidad de los procesos. En la literatura clásica (Barto, 1990; Werbos, 1990; Hunt *et al.*, 1992) se describen tres aplicaciones fundamentales de las redes neuronales a los sistemas de control, *control directo*, *control inverso* y *control indirecto*.

4.2.1 Control Directo

La red neuronal se emplea para apoyar la acción del controlador que calcula la actuación necesaria a partir de la señal de error entre la medida y la consigna. La figura 4.4. muestra el punto en el que se implementa la red neuronal.

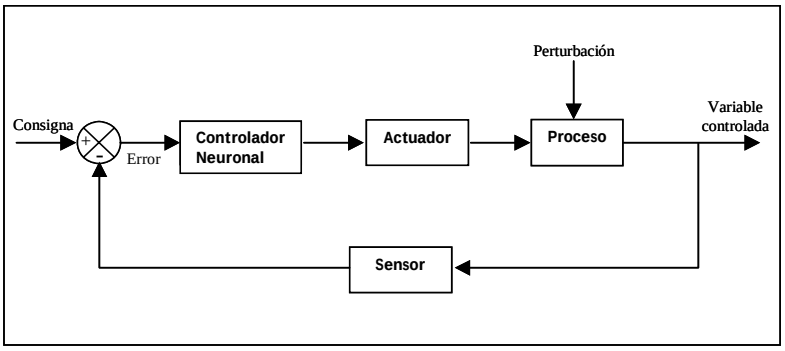

Figura 4.4 Esquema de control directo neuronal.

En realidad en este punto la red funciona como un controlador más, que calcula la actuación adecuada en cada momento. Cómo entrenar la red para que aprenda las actuaciones adecuadas es un problema que puede ser abordado desde diversas perspectivas. El quid del problema radica en disponer de una variable que permita el aprendizaje, bien la salida deseada (la actuación óptima) o alguna relacionada con ella. Una opción común consiste en emplear la red neuronal para reproducir cualquier otro sistema de control, como se muestra en la figura 4.5. Diversos autores (Psaltis *et al.*, 1988; Ichikawa *et al.*, 1992), consideran que una configuración apropiada consiste en utilizar como variables de entrada la consigna actual así como las medidas del proceso en los instantes precedentes, y la salida de la red es la actuación que ha de realizarse en el instante siguientes.

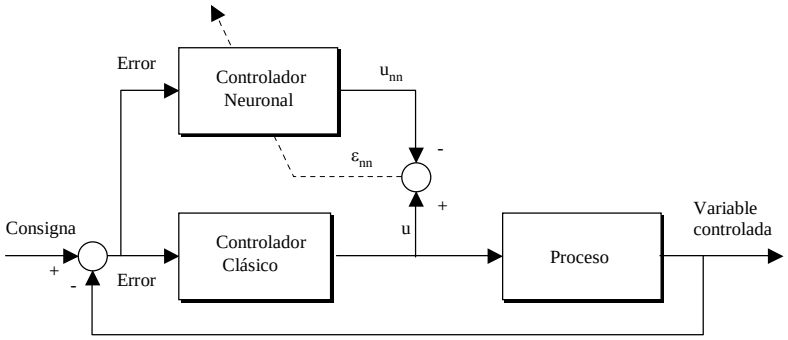

Figura 4.5 Esquema de control directo neuronal, en el que la red se emplea para reproducir un control clásico

4.2.2 Control Inverso

En este caso aprendizaje de la red tiene como objetivo obtener un *modelo inverso* del sistema que se desea controlar. A partir de los valores de la variable controlada (que es normalmente la salida del sistema), y la evolución pasada del sistema, se obtienen los valores de la variable manipulable (entrada) necesaria para que el sistema obtenga la salida deseada.

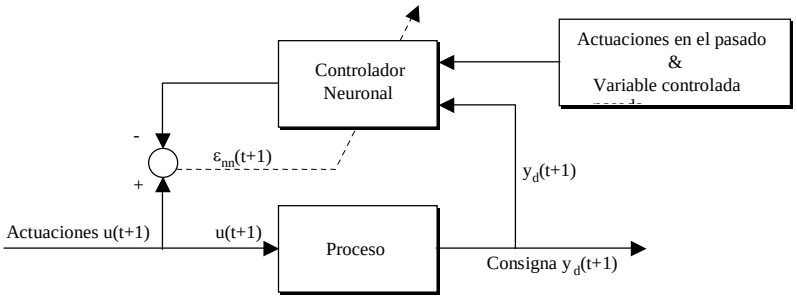

Figura 4.6 Esquema de control inverso

Un modelo del sistema clásico es lo que se denomina un sistema directo, y busca reproducir el comportamiento del sistema, de modo que calcula el valor de la variable de control frente a una acción de control realizada con la variable manipulada, empleando así mismo la información pasada del sistema.

$$\hat{y}_{t+1} = F_1\left(y_t, y_{t-1}, ..., y_{t-a}, u_{t+1}, , u_t, u_{t-1}, ..., u_{t-b}\right) \quad (4.\ 4)$$

Siendo *a* y *b* los retardos relevantes de la variable de control y manipulada respectivamente.

En el caso del control inverso lo que se calcula es la acción del controlador para obtener un determinado valor de la variable controlada, empleando la información del pasado. necesaria a partir del pasado del sistema a, generalmente la consigna.

$$\hat{u}_{t+1} = F_2\left(y_{t+1}, y_t, y_{t-1}, ..., y_{t-a}, u_t, u_{t-1}, ..., u_{t-b}\right) \quad (4.\ 5)$$

Hay numerosas aplicaciones de la aplicación del control indirecto (Barto, 1990; Bhat *et al.*, 1990; Hosogi, 1990; Hunt *et al.*, 1992; Psichogios *et al.*, 1991; Nahas *et al.*, 1992).

Se puede construir un controlador que contenga tanto el modelo inverso como el directo estimado con redes neuronales. El modelo inverso sirve para controlar el sistema, mientras que el directo evalúa la concordancia entre la salida real y la predicha por el modelo directo neuronal.

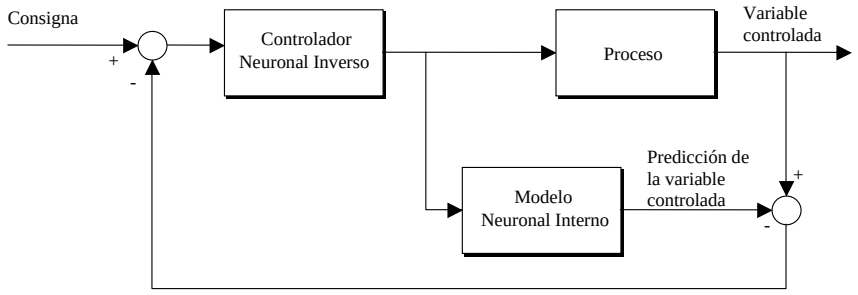

Figura 4.7 Esquema de control inverso que invierte un modelo neuronal del sistema

Estos modelos requieren un estudio detallado de las dependencias temporales, puesto que pueden surgir problemas en aquellos que presentan retardos temporales amplios (Stephanopoulos *et al.*, 1996; Shinskey, 1994).

4.2.3 Control Indirecto

En este caso se emplea la red neuronal para obtener el modelo del sistema, o bien para determinar las consignas de los posibles lazos de control. El caso mostrado en la figura 4.7 podría encuadrarse también en lo que se llama un modelo indirecto – inverso; se encuentra de hecho a caballo entre un control inverso y un control indirecto. La red en este caso actúa como un modelo que ayuda al sistema, bien a establecer las consignas adecuadas, bien a predecir el valor de la variable en un instante futuro. La definición de control indirecto es tan amplia que casi todo aquello que no pueda ser definido como directo o inverso se engloba dentro del control indirecto.

Muchos de los procesos que requieren un control exhaustivo están sujetos a importantes cambios del sistema. Por esto los modelos de control *adaptativos*, esto es, que evolucionan siguiendo al sistema, son en general más efectivos que los *no adaptativos*. Es por ello que se hace necesario un reentrenamiento frecuente de la red neuronal, pero sin dejar de considerar la necesidad de tener un conjunto representativo de datos como base. Algunos trabajos evidencian problemas a este respecto (Steyer *et al.*, 2000), debido a un excesivo reentrenamiento, asociado a una selección incorrecta del conjunto de entrenamiento. Resulta habitual en la literatura escoger ventanas móviles como conjunto de entrenamiento, pero esto, en el caso de sistemas controlados resulta erróneo, pues si el sistema se encuentra bajo control la información de la que se dispone es muy escasa, en muchos casos es prácticamente ruido, y es por ello que ante perturbaciones del sistema más severas el controlador no actuará correctamente, pues no se han considerado datos relevantes en su entrenamiento.

En esta tesis se van a considerar dos nuevos enfoques en la aplicación de las redes neuronales al control de procesos, fundamentalmente basados en el control indirecto del sistema. Las metodologías empleadas van a ser fundamentalmente dos, la primera enfocando las redes como parte de un control on/off, con el fin de detectar cuando el sistema se encuentra en

riesgo, y en el segundo como parte de un controlador continuo, con el fin de mantener el sistema en el rango de funcionamiento deseado.

4.3. Redes Neuronales en Procesos de Control. Predicciones Temporales.

En muchos procesos de control resulta inviable la actualización en un único intervalo de tiempo de la variable controlada al valor de consigna. Es por ello que en ocasiones se desea que se alcance la consigna en un horizonte temporal, que depende de las características del proceso y de la suavidad con que se desee controlar el proceso. Bajo esta filosofía la variable de control que se desea considerar no es el siguiente paso de la variable de control, sino un retardo posterior. Este retardo ha de ser predicho, por lo que se pueden emplear diferentes metodologías estadísticas para modelizar las series temporales.

La metodología será pues introducir la predicción de la variable controlada en el proceso de control ya existente, con el fin de suavizar la actuación del controlador al tiempo que permite la toma anticipada de acciones. La figura 4.8 muestra el esquema del control.

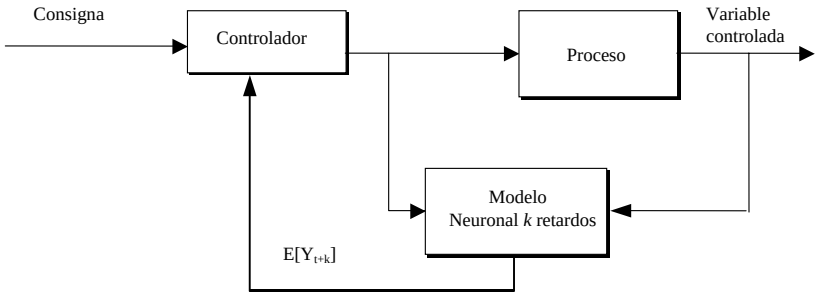

Figura 4.8 Esquema de control indirecto con predicción temporal

La acción del controlador se encamina ahora a obtener la consigna de control en *k* retardos, no de modo inmediato, lo que al tiempo que hace además de anticiparse a ciertas variaciones previsibles de la variable de control, puede ayudar a disminuir las oscilaciones del controlador en torno a la consigna. La bondad del controlador estará pues ligada a la bondad del predictor y a la libertad que presente el controlador en la toma de decisiones. Si el proceso es muy suave, y el horizonte temporal no es muy grande, las diferencias entre las acciones de control del controlador clásico y aquel que emplea las predicciones no serán muy acusadas; si el proceso es rápido, y de cambios bruscos, se limita la amplitud del futuro a predecir, pero si las predicciones son adecuadas se mejora significativamente la acción de control.

En el proceso de desarrollo de un controlador resulta imprescindible considerar las características propias de cada proceso, referentes tanto a la naturaleza del mismo como a las características técnicas e instrumentales del mismo. De este modo un factor a tener en cuenta es el tipo de variable manipulable que se considera, asociado al tipo de actuador disponible. Muchos procesos, como el control de la temperatura de un refrigerador, presentan actuadores binarios (On/Off), mientras que otros, como el control de un horno a través de la altura del electrodo, operan con actuadores continuos.

Para ilustrar estos controladores indirectos con predicción temporal se emplearán dos ejemplos diferentes, uno en el que la acción del controlador es binaria, y otro en el que es continua.

El primero se refiere a un proceso industrial de colada de silicio en placa de cobre, y el segundo está ligado al tratamiento anaeróbico de aguas residuales.

4.3.1. Control de Colada de Cobre

En este ejemplo se muestra la aplicación de redes neuronales al control un proceso industrial. En primer lugar se explicará el proceso industrial que se desea controlar, para a continuación explicar la aportación de las redes neuronales al proceso.

4.3.1.1. Colada en Placa de Cobre

El proceso de denomina *Colada en Placa Continua de Cobre* y es un proceso desarrollado por Ferroatlántica I+D que se emplea en la fábrica que Ferroatlántica SL posee en Sabón, (A Coruña).

El silicio metal es un material empleado en la industria para aplicaciones muy diversas, desde la fabricación de siliconas hasta su presencia en aleaciones de aluminio, aunque quizás la aplicación del silicio más en boga en estos tiempos es la fabricación de células de energía solar fotovoltaica. La placa continua de cobre es un procedimiento de colada, que permite la distribución homogénea de impurezas, lo que eleva la calidad del metal obtenido. Así mismo es un proceso más limpio, que no genera residuos.

La figura 4.9 representa el esquema de la placa continua de cobre , sobre el que se explicaré su funcionamiento.

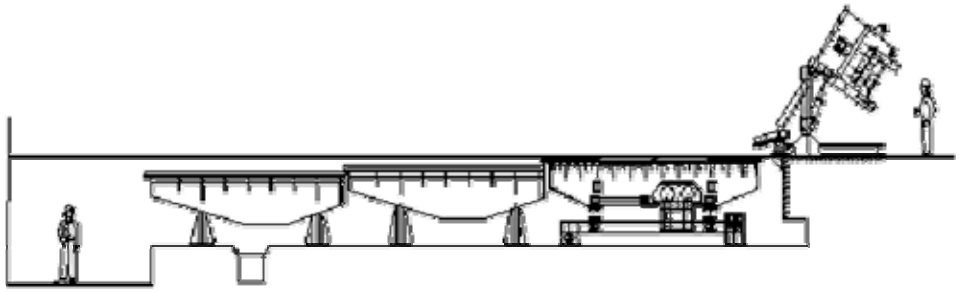

Figura 4.9 Esquema de la colada en placa continua de cobre.

El objetivo es la solidificación del metal incandescente con el fin de obtener un producto uniforme. . La placa de cobre está formada por 3 mesas vibratorias de cobre consecutivas refrigeradas internamente con agua. En el inicio de la estructura se sitúa una cubeta que vierte el metal fundido sobre la primera placa. El vertido del metal se hace oscilando el volcador de derecha a izquierda y viceversa, de modo que se llena la parte inicial y las vibraciones hacen que el metal en proceso de solidificación avance, dejando libre la parte inicial de la primera placa para volver a ser rellenada. El metal se solidifica a lo largo de las 3 placas al ser enfriado gracias a los sistemas de refrigeración por agua que hay bajo las placas.

4.3.1.2. Sistema de control auxiliar. Alarma por temperatura

El control de este proceso industrial tiene como objetivo la solidificación continua del metal, de modo que el metal no esté demasiado tiempo estático en un punto, -lo que podría generar el fundido de la placa con la consecuente fuga de agua bajo el metal, que en estas circunstancias puede provocar explosiones por la presión del agua evaporada bajo el metal – pero que esté en las placas el tiempo suficiente para llegar en estado sólido a la parte final de la estructura. Además la duración de la colada está condicionada por el hecho de que se prolonga demasiado en el tiempo el metal fundido que aguarda en la cubeta para ser vertido puede solidificarse, con la consiguiente pérdida de material y/o de energía si se opta por fundirlo de nuevo antes de volver a ser colado.

La información de la que se dispone para controlar el proceso se obtiene a partir de termopares que miden la temperatura en distintas partes de la estructura de la placa de cobre. La figura 4.10 muestra el esquema de los sensores de temperatura situados a lo largo de la placa de cobre.

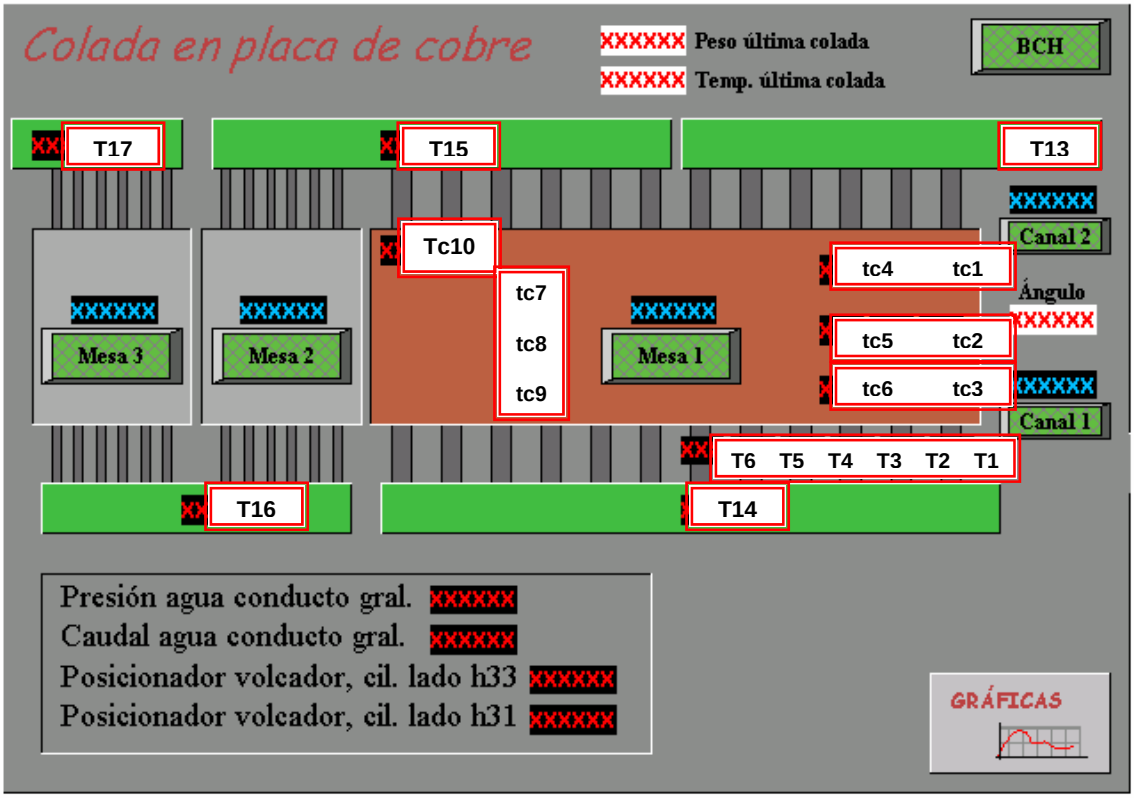

Figura 4.10 Esquema de la colada de cobre.

Inicialmente la Colada en Placa de Cobre se regulaba a través de la temperatura del agua de refrigeración en diferentes puntos Cuando se produce una acumulación de silicio, los termopares iniciales sufren una disminución de temperatura, pues se forma una capa que aísla el termopar y el silicio que sigue cayendo. El aumento de silicio en la placa se refleja entre otras variables en la temperatura del tc10 en forma de aumento de temperatura.

El funcionamiento tradicional se fundamenta en una velocidad de volcado constante basada en la geometría estimada del interior de la cubeta. El sistema de control es binario y consiste

en detener el proceso cuando la temperatura del termopar denominado **tc10** supere un umbral e seguridad, pues constituye un buen indicador de la cantidad de metal que hay en la placa.

4.3.1.3. Predicción de la temperatura con redes neuronales

Se propone un controlador basado en redes neuronales anticipativo, que disminuye la velocidad de volcado cuando la predicción a un horizonte temporal preestablecido de la temperatura del termopar *tc10* alcanza un umbral. La determinación de este umbral está basada en las características de la estructura y la experiencia previa en la operación del sistema.

Los sensores proporcionan datos a intervalos de 5 segundos. El objetivo es predecir la variable **tc10** a un determinado horizonte temporal, con el fin de disminuir la velocidad de volcado del metal y evitar que sea necesario detener el proceso. Esta predicción se podría abordar desde dos puntos de vista: continuo y discreto. Continuo si se desea predecir toda la curva de temperatura, y discreto si sólo se desea predecir con antelación si va la variable va superar un umbral, sin importar el valor que tome. En este caso se ha optado por la primera opción. Se presentarán pues los resultados resultante de predecir la evolución del valor de la variable **tc10**.

Los pasos para el diseño del controlador fueron: la determinación del horizonte temporal a la hora de hacer la predicción, construir el modelo y el establecimiento de sistema de alarma.

El horizonte temporal se estableció en 25 segundos, porque proporcionaba un margen de manobra adecuado, con una predicción adecuada. Se probaron diferentes modelos de predicción, serie de tiempo, regresión dinámica y finalmente redes neuronales. En este documento se presenta sólo la aplicación de redes neuronales. El conjunto de entrenamiento se creó empleando coladas con distintos perfiles, típicas, como la que se muestra en la FIGURA 4.11, y también coladas con incidencias como la formación de capa gorda, aparición de humo derivado de fugas de agua,…

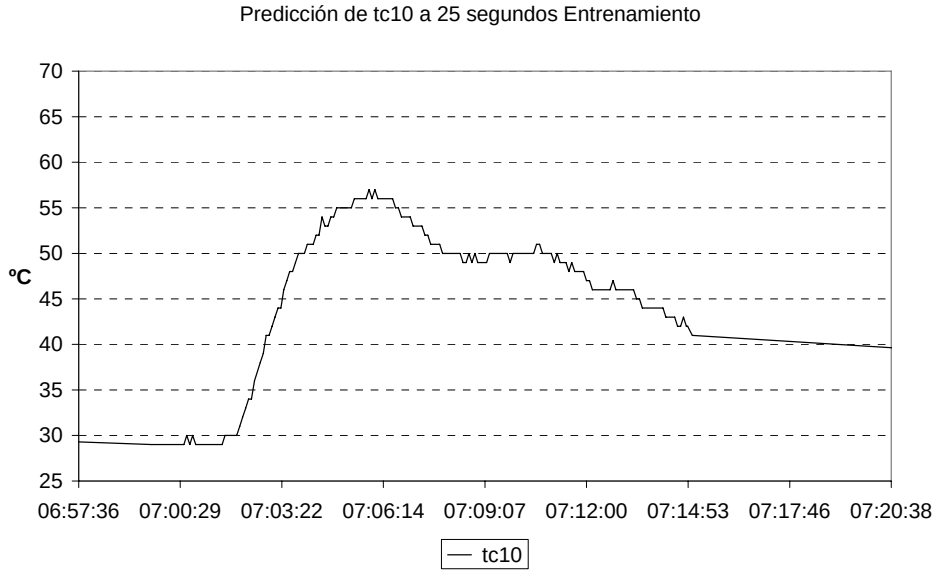

Figura 4.11 Ejemplo de colada.

La elección de las variables de entrada en la red se hizo estudiando las dependencias con las medidas de temperatura situadas entre el principio de la placa y la posición del tc10. Las variables consideradas para la predicción del $tc10_t$ fueron:

- Historia del tc10. Valores 25 y 30 minutos antes de la predicción. Dado que los datos se miden cada 5 minutos se denotará, $tc10_{t-5}$, $tc10_{t-6.}$
- Media de las temperaturas de los termopares de primera línea (1, 2 y 3). El estudio e la dependencia temporal determina la elección del retardo empleado. En este caso se seleccionó el duodécimo retardo (1 minuto):$tcmed123_{t-12}$.
- Media de las temperaturas de los termopares de la segunda línea (4, 5 y 6). En este caso el retardo elegido es el séptimo. (35 minutos) $tcmed456_{t-7}$.
- Ángulo de colada en el momento en que se hace la predicción, esto es, ang_{t-5}
- Tiempo transcurrido desde el inicio de colada en minutos, esto es $5t$.
- El peso inicial de la colada: *carga*.
- La temperatura inicial de la colada $tc10_0$.

Los resultados obtenidos con las serie temporal y la regresión dinámica sugirieron que las redes con capa oculta de función lineal podrían ser adecuadas para esta predicción. La estructura de la red neuronal fue la siguiente:

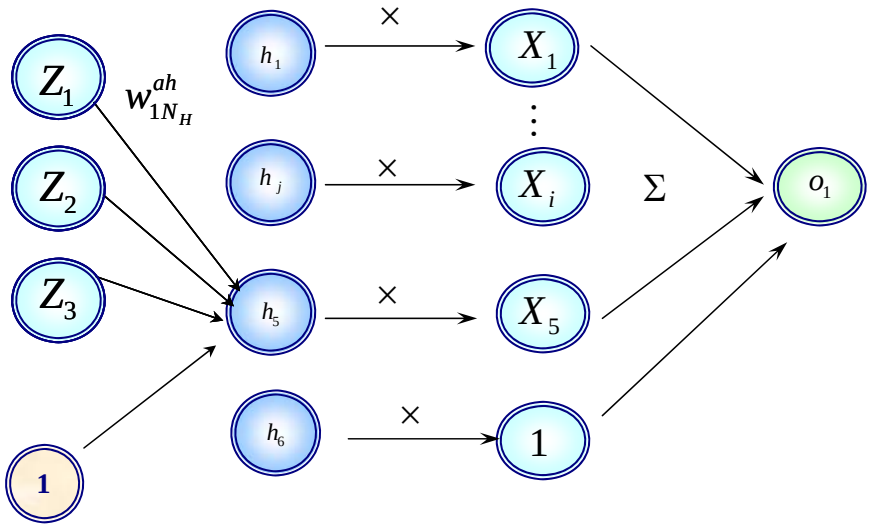

Figura 4.12 Estructura de la red neuronal empleada.

Las variables Z_i son: *carga*, $tc10_0$ y $5t$. Las variables X_i son: ang_{t-5}, $tcmed123_{t-12}$, $tcmed456_{t-7}$, además de retardos 5 y 6 de la variable tc10.

La función de la capa oculta es lineal. De este modo la predicción de la temperatura del termopar 10 en el instante *t* responde a la expresión:

$$\hat{y}_t = o_1 = \sum_{r=1}^{5}\left(\sum_{j=1}^{3} w_{rj}^{ah} Z_j + w_{0r}^{h}\right) X_r + \sum_{j=1}^{3} w_{6j}^{ah} Z_j + w_{06}^{ah} \qquad (4.\ 6)$$

Las siguientes figuras muestran ejemplos de coladas en el entrenamiento y la validación.

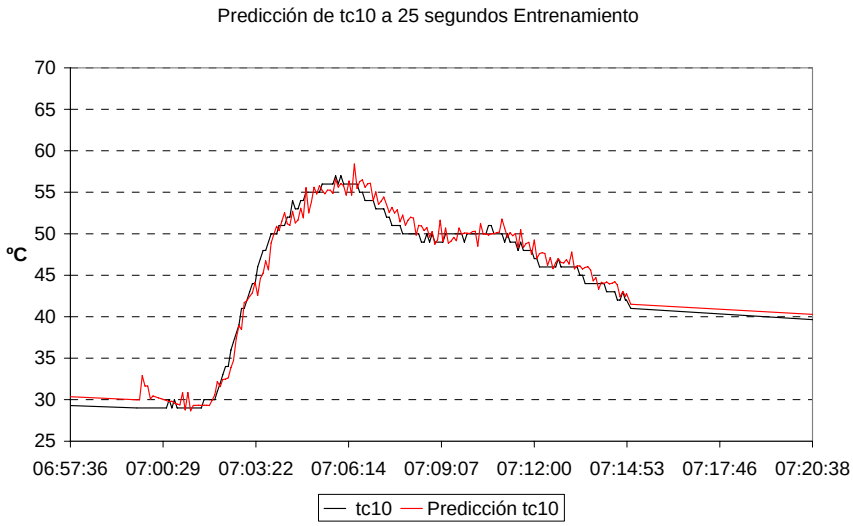

Figura 4.13 Ejemplo de predicción de colada. Entrenamiento.

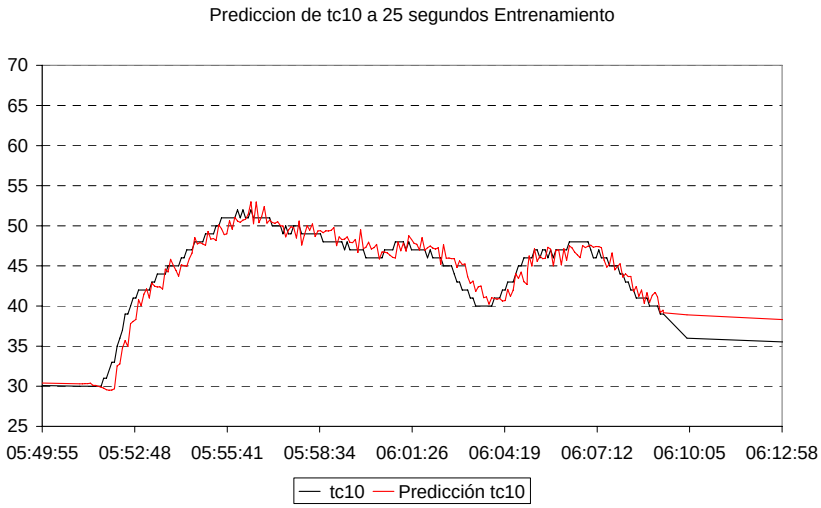

Figura 4.14 Ejemplo de predicción de colada. Entrenamiento.

La variable tc10 es una variable discreta, por lo que a la hora de hacer la predicción en la implementación real se consideró en entero más próximo al valor predicho.

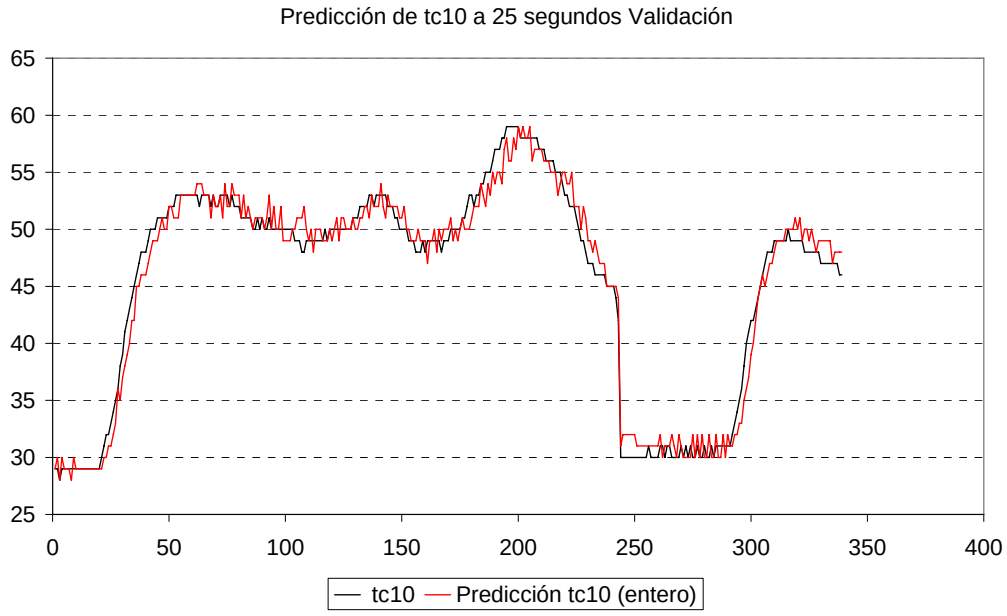

Figura 4.15 Ejemplo de predicción de colada. Validación.

Los errores obtenidos fueron:

		ECM	ERAM	ERM
Predicción Real	**Entrenam.**	3,6732	0,0284	0,0021
	Validación	1,8217	0,0242	0,0006
Predicción Entera	**Entrenam.**	3,7560	0,0279	0,0021
	Validación	1,9086	0,0234	0,0007

Tabla 4.1 Variables online y offline medidas para el seguimiento del proceso.

Siendo ECM el error cuadrático medio, ERAM el error relativo absoluto medio, y ERM el error relativo medio.

Para determinar la alerta de temperatura se respetaron los criterios existentes previamente. Dado que la temperatura ambiente influye considerablemente en la temperatura del agua de la placa de cobre, y que el incremento de temperatura se relaciona con la cantidad de metal en la placa, el umbral de alarmase define en función de la temperatura inicial. Se produce una alerta por subida de temperatura cuando se en un incremento de la temperatura del tc10 de 40 grados centígrados en relación a la temperatura antes del inicio de la colada. Debido al número de alarmas producidas se optó por establecer un número de alarmas necesarios para establecer la bajada de velocidad en el volcado. Serán necesarias dos alarmas consecutivas para la modificación de la velocidad que es una variable escalonada. En cualquier caso se mantiene la alarma previa, esto es, si el valor de la variable *tc10* se incrementa en 40 grados, se detiene la colada.

4.3.2 Control de Una Planta de Tratamiento Anaeróbico de Aguas Residuales

En esta sección se aportarán nociones básicas sobre la digestión anaeróbica, y en particular sobre el tratamiento de aguas residuales en reactores anaeróbicos, que es el proceso para el que se desea elaborar una estrategia de control. Así mismo se detallará el proceso de selección

de las variables de control, así como la construcción del modelo interno predictor temporal de las variables de control, con varios horizontes temporales. Finalmente se tratarán los resultados obtenidos.

4.3.2.1. Introducción a la Digestión Anaeróbica

La Digestión Anaeróbica (AD) es un proceso complejo mediante el cual la materia orgánica se transforma en compuestos más simples sin la necesidad de la participación de oxígeno molecular en el proceso (Switzenbaum, 1995). Los tratamientos anaerobios son procesos complejos en los que tienen lugar un gran número de procesos (de tipo biológico y físico-químico) que se desarrollan tanto en serie y en paralelo, generando un espacio multidimensional de respuestas y operación. El análisis multivariante, por tanto, resulta adecuado para la estudio de este tipo de procesos (Ruiz ***et al.***, 2005a).

En general el proceso AD se puede dividir en 3 etapas, la hidrólisis extracelular, durante la cual se transforman los compuestos orgánicos complejos en azúcares y aminoácidos; la fermentación que a su vez se divide en dos subetapas, la acidogénesis, que transforma estos compuestos ácidos orgánicos y alcoholes, y la *acetogénesis*, en la que los compuestos resultantes de la acidogénesis se descomponen en hidrógeno y ácido acético; y finalmente la metanogénesis que también puede dividirse en dos subetapas diferentes, que son la *metanogénesis hidrogenotrófica* en la que el hidrógeno se transforma en metano, y la *metanogénesis acetoclástica*, en la que el acético se transforma en metano.

Las distintas reacciones presentan velocidades muy diferentes, de modo que la velocidad de producción de los ácidos es mucho más elevada que la velocidad de producción de metano. Es por ello que la aparición de compuestos fácilmente degradables da lugar a un aumento en la generación de ácidos, que no pueden ser transformados en metano a la misma velocidad, lo que genera una acumulación de compuestos intermedios. Las acumulaciones de productos intermedios pueden dar lugar a algún efecto inhibitorio, bien en la reacción que lo genera (como producto) bien en la siguiente (como substrato). La velocidad del proceso considerado de modo global está regida por la velocidad del proceso más lento. Dependiendo del compuesto que se esté degradando las etapas limitantes serán bien la metanogénesis, o bien la hidrólisis. La figura 4.9 muestra el esquema de la digestión anaerobia.

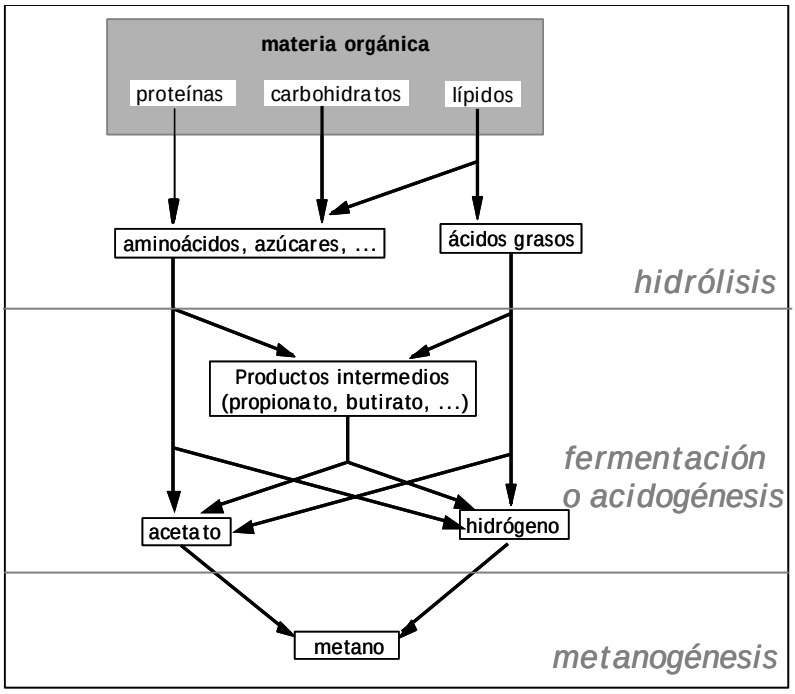

Figura 4.9 Esquema de la digestión anaerobia.

En sus comienzos las aplicaciones de los sistemas de tratamiento anaeróbicos se centraban en la estabilización de residuos de alta carga orgánica, en la mayoría de los casos lodos procedentes de los tanques de sedimentación de plantas de tratamiento de aguas residuales. Esto se debía a que los modelos convencionales estaban limitados por la necesidad de largos tiempos de residencia para la estabilización del residuo, por lo que no se consideraban apropiados ni rentables para el tratamiento directo de aguas residuales (Young y McCarty, 1969). El desarrollo de nuevos reactores capaces de operar con largos tiempos de retención de sólidos (con zonas diseñadas para retener cantidades elevadas de biomasa) y cortos tiempos de retención hidráulica, *THR*, mejoró significativamente la economía del sistema (Switzenbaum *et al.*, 1990) y originó su expansión competitiva a múltiples áreas. comenzó a ser más utilizada cuando se logró el diseño de reactores anaerobios

A lo largo de los últimos años, los procesos de tratamiento de aguas residuales en digestores anaerobios han sido aplicados con éxito en multitud de situaciones; aunque predominantemente se emplea para el tratamiento de efluentes especialmente de origen industrial, también se ha aplicado de forma eficaz en el tratamiento de aguas residuales urbanas (Switzenbaum, 1995; Hulshoff Pol *et al.*, 1997).

Las principales ventajas que presenta el tratamiento anaerobio de aguas residuales son fundamentalmente las bajas demandas que requiere, tanto energéticas como de nutrientes, así como la baja producción de lodos (Rintala, 1991), unidos a la capacidad de tratar sustratos poco biodegradables en altas concentraciones, y la posibilidad de recuperar energía a través de la producción de metano, y en menor medida de hidrógeno. Presenta también ciertas limitaciones que han de ser tenidas en cuenta como la lentitud del proceso de arranque derivada del lento crecimiento de lo microorganismos, que puede ser suplida por la inyección de lodo activo de otros reactores en funcionamiento, la sensibilidad de los microorganismos a condiciones adversas o la necesidad ocasional de postratamiento.

Para los distintos estudios desarrollados en esta tesis se emplearon datos de una planta piloto. El reactor es un híbrido UASB-UAF de aproximadamente 1 m^3 de volumen útil, mostrada en figura 4.10 (Fernández, 1994; Fernández *et al.*, 1995). La planta esta completamente instrumentada de modo que cuenta con medidores de flujo de alimentación, y recirculación, medidor de pH, medidores de flujo de gas, analizador de gases con infrarrojos (metano y dióxido de carbono), analizador de hidrógeno y analizador por combustión de TOC/TIC. Se pueden además calcular algunas variables de interés el flujo de metano, de hidrógeno, o la velocidad de carga orgánica. La tabla 4.2 muestra las variables de las que se dispone para el seguimiento del proceso.

	Variable	Abreviatura	Medidas	Unidades
1	Caudal de Alimentación	Qa	On-line	L/h
2	Temperatura del influente	Tin	On-line	ºC
3	Temperatura del reactor	Tr	On-line	ºC
4	Flujo de Recirculación	Qr	On-line	L/h
5	Presión en la cabeza del reactor	P	On-line	Mbar
6	Concentración de metano en la fase gas	$\%CH_4$	On-line	%
7	Contenido de hidrógeno en la fase gas	H_2	On-line	Ppm
8	PH del efluente o en el reactor	pH eff	On-line	
9	Carbono orgánico disuelto en el efluente	DOC eff	On-line	mgC/L
10	Carbono inorgánico disuelto en el efluente	DIC eff	On-line	mgC/L
11	Carbono orgánico total en el efluente	TOC eff	On-line	mgC/L
12	Carbono inorgánico total en el efluente	TIC eff	On-line	mgC/L
13	Carbono orgánico total en el influente	TOC inf	On-line	mgC/L
14	Carbono inorgánico total en el influente	TIC inf	On-line	mgC/L
15	Caudal de gas	Qgas	On-line	L/h
16	Caudal de metano	QCH_4	On-line	L/h
17	Velocidad de carga orgánica	OLR	On-line	$kgCOD/m^3 \cdot d$
18	PH del influente	pH in	On-line	
19	Concentración de etanol en el influente	EtOH inf	Off-line	g/L
20	Concentración de acetato en el influente	Acet inf	Off-line	g/L
21	Concentración de propionato en el influente	Prop inf	Off-line	g/L
22	Concentración de butirato en el influente	But inf	Off-line	g/L
23	Concentración de etanol en el efluente	EtOH eff	Off-line	g/L
24	Concentración de acetato en el efluente	Acet eff	Off-line	g/L
25	Concentración de propionato en el efluente	Prop eff	Off-line	g/L
26	Caudal de hidrógeno	QH_2	On-line	g/L

Tabla 4.2 Variables online y offline medidas para el seguimiento del proceso

Figura 4.10 Imagen del reactor UASB-UAF.

4.3.2.2 Necesidad de un sistema de monitorización y control en un reactor anaeróbico

La complejidad del proceso y su necesidad de estabilidad hacen necesario un sistema de monitorización y control. Uno de las mayores dificultades es la obtención de un estado estable que impida la acumulación de productos intermedios, como ácidos grasos volátiles (Pullammanappallil et al., 2001). Estas acumulaciones son debidas a los cambios a los que está sujeto el influente, tanto de cantidad como de calidad, y a la presencia esporádica de tóxicos, que a menudo se producen en las plantas de tratamiento de aguas industriales, debido a la que los efluentes dependen de los ciclos de producción, que no suelen ser estables ni homogéneos, transmitiendo esta variabilidad al efluente.

Entre los beneficios que proporciona un sistema de control destacan (Olsson y Newell, 1999) la *mejora de la calidad del efluente*, que en estos días es tan importante para evitar vertidos altamente contaminadas; *la economía del proceso*, que permite el uso adecuado de los recursos, tanto energéticos como los asociados a los nutrientes de modo que tanto el sobredimensionamiento como los márgenes pueden ser menores; *la mejora de la operabilidad de un sistema complejo*, que hace que no sean necesarios operarios fuertemente especializados para el manejo del sistema Se han planteado así mismo algunos inconvenientes entre los que se pueden mencionar *la necesidad de una inversión económica*, pues aunque a largo plazo los sistemas de monitorización y control generan mejoras económicas, es necesaria una inversión inicial que algunos industriales ven como un coste de operación adicional, que no desean realizar; la *falta de sensores en el marcado*, adecuados para el seguimiento de muchas de las variables del sistema, de modo que los existentes son complejos, requieren un operador experto y paradas periódicas por mantenimiento.

Entre los objetivos que tiene el sistema de control destacan: la *eliminación de perturbaciones*, definiéndolas como las que no dependen del operador y son producto de las variaciones de las condiciones ambientales: variaciones del caudal, temperatura del influente, temperatura ambiental, presencia de tóxicos, etc. El sistema de control tendrá como fin disminuir o anular los efectos negativos que puedan tener esas variaciones no deseadas en el proceso; la *estabilidad del proceso*, pues el disponer de un sistema de control que consiga mantener en un rango adecuado las variables ambientales y operacionales hace que el sistema sea más estable; la *optimización del rendimiento*, de modo que varia las condiciones operacionales para mantener el proceso con un rendimiento adecuado.

4.3.2.3 Selección de Variables

Para desarrollar un algoritmo de control es necesario en primer lugar seleccionar las variables que van a actuar como variable controlada y variable manipulable; la primera cuestión es determinar la variable controlada; esta variable ha de ser adecuada tanto desde un punto de vista práctico (que el sensor exista a un precio razonable en el mercado y que dicha variable efectivamente se pueda medir) y desde un punto de vista teórico (que la variable tenga algún significado metabólico, que posea un tiempo de respuesta pequeño, etc) (Ruiz *et al.*, 2002). Dicha variable deberá proporcionar información sobre el estado del sistema y, al tiempo, ser sensible a los cambios que se produzcan en las condiciones del proceso. La variable manipulable ha de estar relacionada con la variable controlada y al tiempo ha de cumplir que

el operador pueda manipularla; la variable más adecuada suele ser el caudal de alimentación, puesto que regula en gran medida el estado del sistema.

Se han desarrollado diversos estudios bajo distintos sustratos con el fin de establecer el número mínimo de variables que es necesario monitorizar para controlar un sistema de estas características; para ello se ha empleado el Análisis Factorial Discriminante, FDA, para la determinación más adecuada de las variables a monitorizar (Molina *et al.*, 2009; Castellano *et al.*, 2007; Ruiz, 2005) basándose en aquellas que daban más información sobre los distintos estados estacionarios del sistema, en términos de identificación de los diferentes estados estacionarios. El trabajo determinó que la posible selección de las variables controlables no es única, sino que existen diversas combinaciones de variables del sistema que permiten distinguir entre las distintas situaciones del proceso; razones económicas, técnicas y de dinámica del sistema llevaron a seleccionar al hidrógeno, H_2, como una variable clave en el funcionamiento del sistema, y por tanto como la variable controlada. En primer lugar esta variable tiene influencia importante en la descomposición de componentes principales, y así mismo clasifica adecuadamente mediante FDA los distintos estados estacionarios de ciertos experimentos realizados; así mismo existen sensores adecuados en el mercado para su medición, económicos, y así mismo es una variable correspondiente a la fase gas, lo que hace que sea de respuesta rápida en comparación con las de la fase líquida. Así mismo el hidrógeno está relacionado con el DOC (demanda química de oxígeno) en el efluente de modo que la consigna del hidrógeno se relaciona con la calidad del efluente (Ruiz, 2005). El metano también es una variable adecuada para el estudio salvo por la falta de biyectividad del proceso debido a los problemas por inhibición que presentan las bacterias metanogénicas. Los distintos substratos empleados dieron lugar a resultados diferentes, con la aparición de términos asociados a la fase líquida del proceso (Molina *et al.*, 2009), si bien tanto el hidrógeno como el metano aparecen como significativos en todos ellos.

4.3.2.4 Modelo de Control existente

Como se señaló en su momento en este punto la aportación de las redes neuronales al controlador se va a centrar en el uso del control de procesos sobre la variable predicha a cierto horizonte de control, no en el estado actual; el objetivo de este proceso es suavizar el control y evitar fluctuaciones, además de seguir la filosofía de que en la práctica no es posible devolver el estado a la consigna en un solo instante temporal, sino que es necesario más tiempo, tiempo que proporciona la predicción.

Se empleará como base un control presente en Rodriguez, (2006) en el que se consideraba el hidrógeno como la variable a controlar, de modo que en la función que calcula la acción de control se tienen en cuenta los valores del metano. Esto es debido a que la acumulación de elementos intermedios genera una inhibición de las bacterias metanogénicas, de tal manera que inestabiliza el sistema. Para evitar esta desestabilización, y considerando un modelo de Haldane, se ha estimado (Ruiz, 2005b) el caudal máximo de metano sin inhibición a partir de la relación entre el caudal de metano y el carbono orgánico disuelto. Así mismo se ha empleado Wild Bootstrap para crear intervalos de confianza para los parámetros del modelo. El objetivo de la estimación de este modelo es evitar que el sistema entrase en zona de inhibición.

Si se denota por ***D*** el caudal de alimentación, la forma del controlador sigue (4. 7)

$$D(t) = D(t-1) + \Delta t \cdot D \cdot K \cdot f_{QCH_4}(QCH_4) \cdot f_{H_2}(H_2) \qquad (4.\ 7)$$

El objetivo es determinar la acción del controlador con respecto al instante anterior. Los criterios que rigen el controlador son los siguientes: si el valor del está bajo la consigna, H_2^*, el valor de la función f_{H_2} será positivo, mientras que si el valor está por encima de la consigna tomará un valor negativo. De esta forma la concentración de hidrógeno en fase de gas influirá en el sentido de la acción, del cambio del caudal de alimentación (disminución o aumento). Por otro lado, el valor de f_{QCH_4} será siempre positivo, y su magnitud será menor cuanto más se aproxime a un valor máximo de productividad de metano del reactor, denominado $QCH_4 \max$. Con esta estructura la función del metano será influir en la magnitud de la acción. Los valores de ambas funciones tendrán un valor absoluto menor o igual a 1.

De este modo, cuando el reactor esta subcargado, tanto la función del hidrógeno como la del metano estarán cerca de 1, por lo que el controlador aumentará el caudal rápidamente, con un incremento K. A medida que el metano aumenta, el valor de su función disminuirá, frenando el aumento del caudal, con el fin de mantenerse próximo a su nivel de productividad máximo. Si se supera el valor de consigna del hidrógeno la acción del controlador será la de disminuir el caudal de alimentación hasta llegar a un valor de equilibrio.

Ante una sobrecarga, los niveles de metano e hidrógeno aumentarán de modo que, cuando el hidrógeno supere la consigna el caudal de alimentación disminuirá, debido a que la función asociada al hidrógeno tomará un valor negativo. Si la sobrecarga es considerablemente grande, llegando a inhibir el proceso, se producirá una disminución en el caudal de metano, lo que provocará que su función asociada se aproxime a 1, incrementando por su parte un aumento de la velocidad de disminución del caudal de alimentación.

A mayores de lo señalado se consideró un sistema para evitar la inactividad del controlador cuando el valor del metano coincida con $QCH_4 \max$. Es el denominado *factor de empuje,* α, que es el valor que tomará la función asociada al metano en ese caso. De este modo aún cuando el reactor esté operando en su máximo nivel de metano, el controlador seguirá funcionado e intentará si es el caso aumentar el caudal de alimentación. De este modo se puede tener en cuenta cierta adaptación de la biomasa y, por lo tanto, el posible aumento de la actividad y de la capacidad productora de metano del reactor.

Una vez establecidos los limites y el comportamiento de las funciones, faltaba por fijar la forma de la función. Para corregir las subcargas las subidas han de ser suaves (de modo que f_{H_2} debe ser una función que baje suavemente de 1 a 0 cuando el hidrógeno sea menor que la consigna) y para corregir las sobrecargas las respuestas han ser más rápidas.

La elección de funciones no es única. Las elegidas fueron.

$$f_{H_2}\left(H_2\right)=\begin{cases}\left(\dfrac{H_2^*-H_2}{H_2^*}\right)^{1/m} & \text{si } H_2 \le H_2^* \\ \left(\dfrac{H_2^*}{H_2}\right)^{n}-1 & \text{si } H_2 > H_2^*\end{cases} \text{, con } n,m \in \mathbb{N} \qquad (4.\ 8)$$

$$f_{CH4}=\frac{\alpha \cdot Q_{CH4}^*}{Q_{CH4}+\alpha \cdot Q_{CH4}^*} \text{ , con } \alpha \in \mathbb{R}^+ \qquad (4.\ 9)$$

En el desarrollo del controlador (Ruiz, 2005) consideró para su sintonización Se consideraron valores de m y n de 2 y 10, respectivamente ya que dan una respuesta suave para valores menores a la consigna de H2 y más brusca para valores superiores. El valor del intervalo de actuación se fijó en 15 minutos, de modo que coincidiese con el intervalo de adquisición del programa de monitorización.

El valor de K se seleccionó de modo que el máximo cambio de caudal fuese del 10% en un intervalo de actualización del controlador, esto es 15 minutos, esto es K=0,4 (h-1). El factor de empuje fue de 0,01. Por razones de seguridad el $QCH_4 \max$ empleado fue el 90% del caudal máximo real del reactor. Ese valor, que tiene un significado físico se calcula para una cinética del tipo Haldane según la siguiente expresión

$$Max\,CH_4 Observado=\frac{q_{CH4}^{\max}}{1+2\sqrt{K_S/K_I}} \qquad (4.\ 10)$$

Siendo $q_{CH4}^{\max}$, K_S y K_I los parámetros de un modelo Haldane, (Ruiz, 2005b), que toman los valores

$$q_{CH4}^{\max}, =1.324\left(\mathrm{L/h}\right), K_S=154\left(\mathrm{mg/L}\right) \text{ , } K_I=231\left(\mathrm{mg/L}\right).$$

Según estas especificaciones el caudal de metano máximo observable es de 503 (L/h) , por lo que, recordando que se desea estudiar el 90% de ese valor como parámetro para el controlador, $QCH_4 \max$ será 452 (L/h)

4.3.2.5. Predicción neuronal de las variables de control

Como se ha señalado se emplea una red neuronal para crear un controlador predictivo o anticipativo. El objetivo es la predicción del comportamiento de las variables de control hidrógeno y metano. El horizonte temporal de predicción debía ser lo suficientemente amplio para proporcionar margen para la actuación del controlador, al tiempo que la magnitud del error no debía ser tan grande que distorsionase la magnitud de la acción de control. Se optó por redes con varios nodos en la capa de salida, de modo que la red minimizase el error a lo largo de un intervalo temporal.

En ambos casos se emplearon perceptrones con una capa oculta para la predicción del metano y del hidrógeno. La función empleada en la capa oculta, como se comentó en aplicaciones

anteriores es una función de tipo logístico. En este caso se ha empleado la tangente hiperbólica. La función en la capa de salida fue lineal. Las redes se diseñaron con el objetivo de predecir la curva de la hora siguiente de modo completo, por lo que se diseñaron con 4 nodos en la capa de salida, de modo que en el entrenamiento se minimizase el error respecto a los datos disponibles en esa hora, esto es datos a t+15 t+30 t+45 y t+60.

Las redes empleadas responden a la siguiente expresión.

$$o_k = \sum_{j=1}^{N_{H^1}} \omega_{jk}^{ho} \cdot f_{h^1}\left(\omega_{0j}^{ah_1} + \sum_{i=1}^{N_I} \omega_{1j}^{ah_1} \cdot X_i \right) + \omega_{0k}^{h^1 o}, \qquad (4.\ 11)$$

$$con\ \ i = 1, \cdots, N_I,\ \ j = 1, \cdots, N_{H^1},\ k = 1, \cdots, N_O\ ,\ N_O = 4\ \text{ y } f_{h^1}(z) = \frac{2}{1+e^{-2z}} - 1,\ z \in \mathbb{R}$$

Predicción del hidrógeno

Como se señaló anteriormente se dispone de datos cada 15 minutos. Sea $H2_t$ el valor del hidrógeno, medido en partes por millón, en el instante t.

El estudio de la dependencia temporal llevó a la elección de los valores de los cuatro retardos anteriores de hidrógeno como las variables de entrada de la red. De este modo, las variables de entrada son: $X_i = \mathrm{H2}_{t-i+1},\ \text{con } i = 1,...,4$

Mientras las variables objetivo son $Y_k = \mathrm{H2}_{t+k},\ \text{con k} = 1,...,4$

Las salidas de la red serán,

$$o_k = \sum_{j=1}^{N_{H^1}} \omega_{jk}^{ho} \cdot f_{h^1}\left(\omega_{0j}^{ah_1} + \sum_{i=1}^{4} \omega_{1j}^{ah_1} \cdot X_i \right) + \omega_{0k}^{h^1 o}, con\ 1 \le k \le N_O = 4 \qquad (4.\ 12)$$

El número de nodos de la capa oculta se corresponden con la red que mejores resultados presenta con el menor número de nodos. En este caso se seleccionó $N_{H^1} = 26$

Predicción del metano

Sea $QCH4_t$ el valor del caudal de metano, medido litros por hora, en el instante t.

El estudio de la dependencia temporal llevó a la elección de los valores de los dos retardos anteriores de metano como las variables de entrada de la red. De este modo, las variables de entrada son: $X_i = QCH4_{t-i+1},\ \text{con } i = 1,...,2$

Mientras las variables objetivo son $Y_k = QCH4_{t+k},\ \text{con k} = 1,...,4$

Las salidas de la red serán,

$$o_k = \sum_{j=1}^{N_{H^1}} \omega_{jk}^{ho} \cdot f_{h^1}\left(\omega_{0j}^{ah_1} + \sum_{i=1}^{4} \omega_{1j}^{ah_1} \cdot X_i \right) + \omega_{0k}^{h^1 o}, con\ 1 \le k \le N_O = 4 \qquad (4.\ 13)$$

El número de nodos de la capa oculta de la red seleccionada fue $N_{H^1} = 18$

El conjunto de entrenamiento empleado está formado por 5 experimentos diferentes de sobrecargas con distintos niveles, desde un 25% de aumento de la concentración de contaminantes en la alimentación (DQO) hasta un 500% de aumento.

La figura 4.11 muestra el perfil del comportamiento de las variables hidrógeno y caudal de metano en el conjunto de entrenamiento.

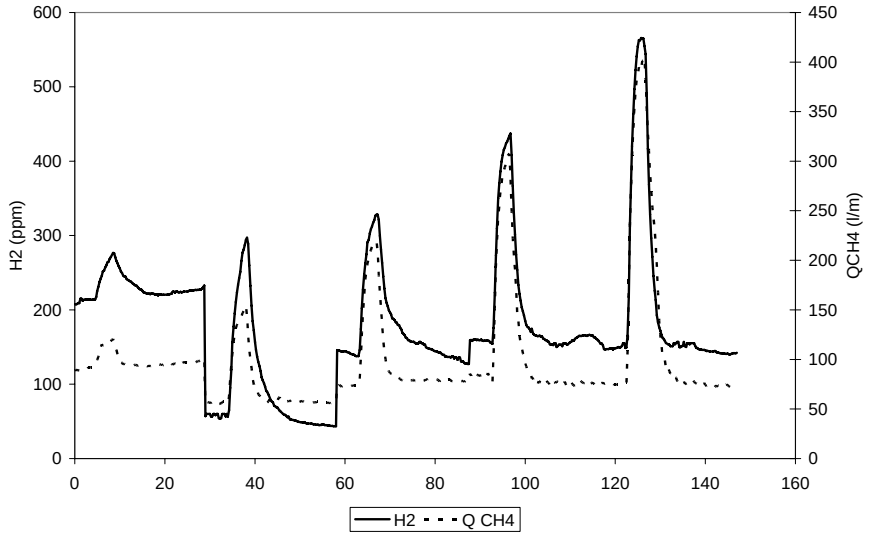

Figura 4.11 Sobrecargas del conjunto de entrenamiento.

Se realizó la validación de su funcionamiento dentro de un lazo cerrado. La figura 4.12 muestra el perfil del experimento

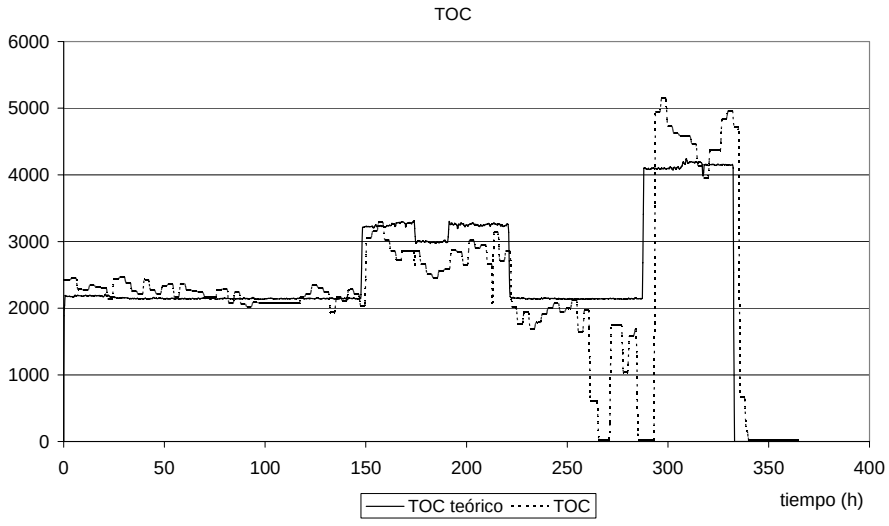

Figura 4.12 Sobrecargas del experimento de validación.

Se llevaron a cabo dos sobrecargas consecutivas. La consigna objetivo era de un nivel de hidrógeno de 25 ppm

Los errores cuadrático medio (ECM) y relativo medio (ERM) de las predicciones durante el experimento de control se recogen en la 3tabla 4.2.

	H2	QCH4
ECM	2,43	5,23
ERM	0,64%	0,42%

Tabla 4.3 Errores de predicción en la implantación del controlador en lazo cerrado.

Muestras del comportamiento de las predicciones de hidrógeno y caudal de metano se muestran en las figuras 4.13 y 4.14., respectivamente.

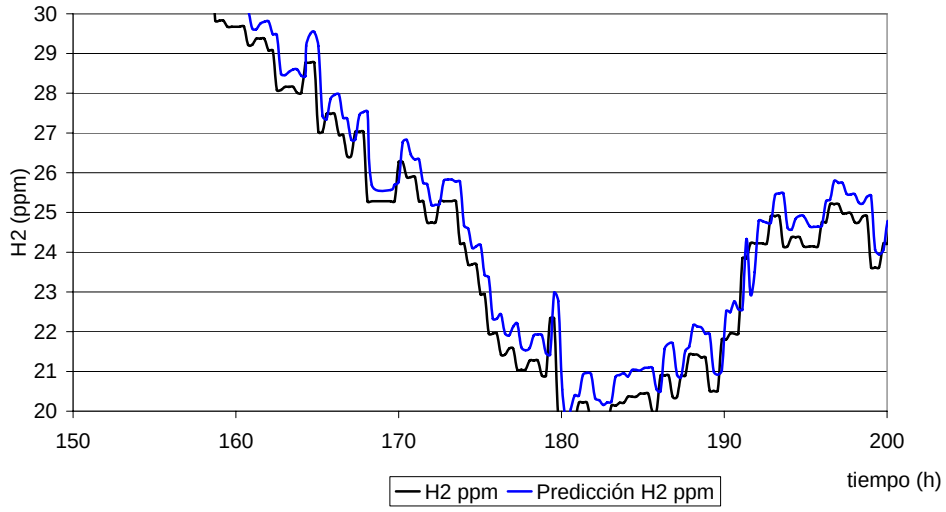

Figura 4.13 Sección del conjunto de validación. Hidrógeno.

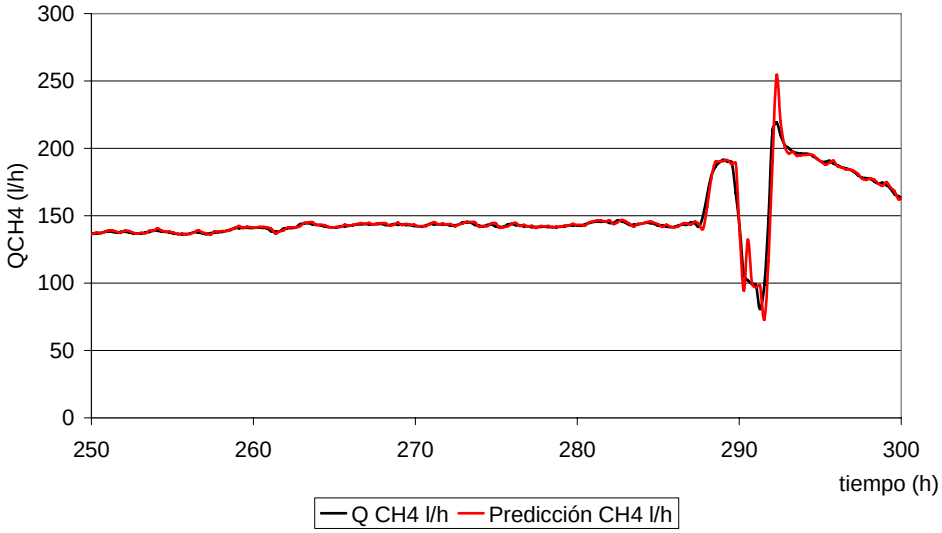

Figura 4.14 Sección del conjunto de validación. Caudal de metano.

Las predicciones no buscan la exactitud, pues el controlador intenta reajustar el hidrógeno hacia la consigna, modificando el curso del caudal. No pasa lo mismo con el caudal de metano, pues no se busca su ajuste, por lo que la predicción de esta variable resulta ser más exacta.

El comportamiento del controlador fue el siguiente.

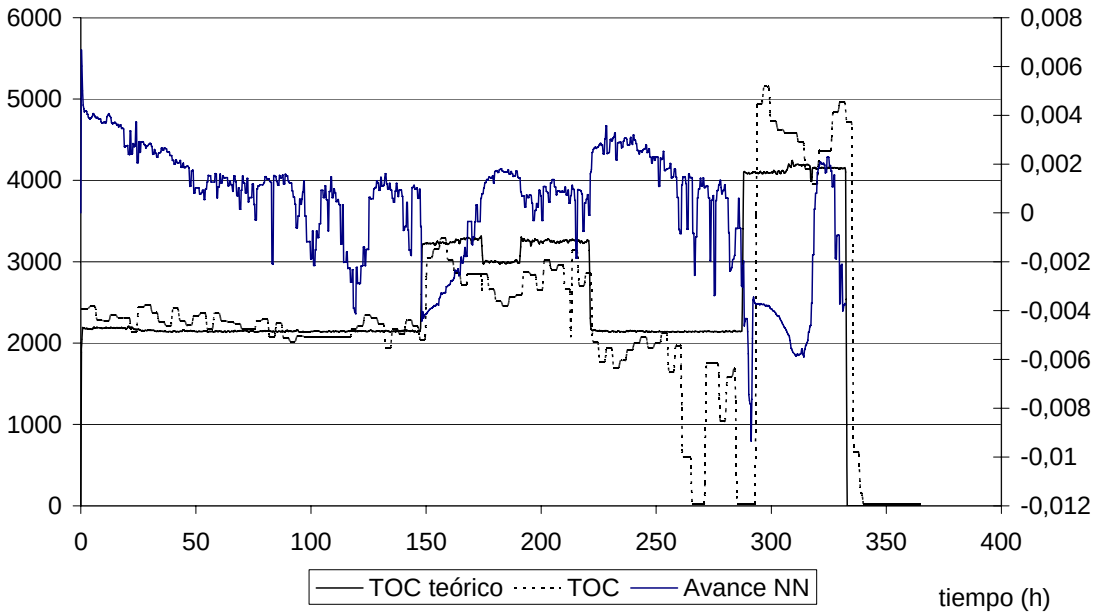

Figura 4.15 Acción de corrección del controlador.

Debido a que el controlador funcionaba en lazo cerrado no es posible saber cómo se hubiese comportado el controlador sin la predicción de la red neuronal. Se puede considerar el lazo abierto y estudiar la posible actuación del controlador, teniendo presente que la evolución del sistema no responde a las acciones del controlador inicial, sino a las del controlador que operaba con NN. Tan solo es posible, pues, observar los puntos en los que se producen claras desviaciones y analizar las posibles causas. Se muestra la comparativa en una sección del conjunto de validación en la figura 4.16.

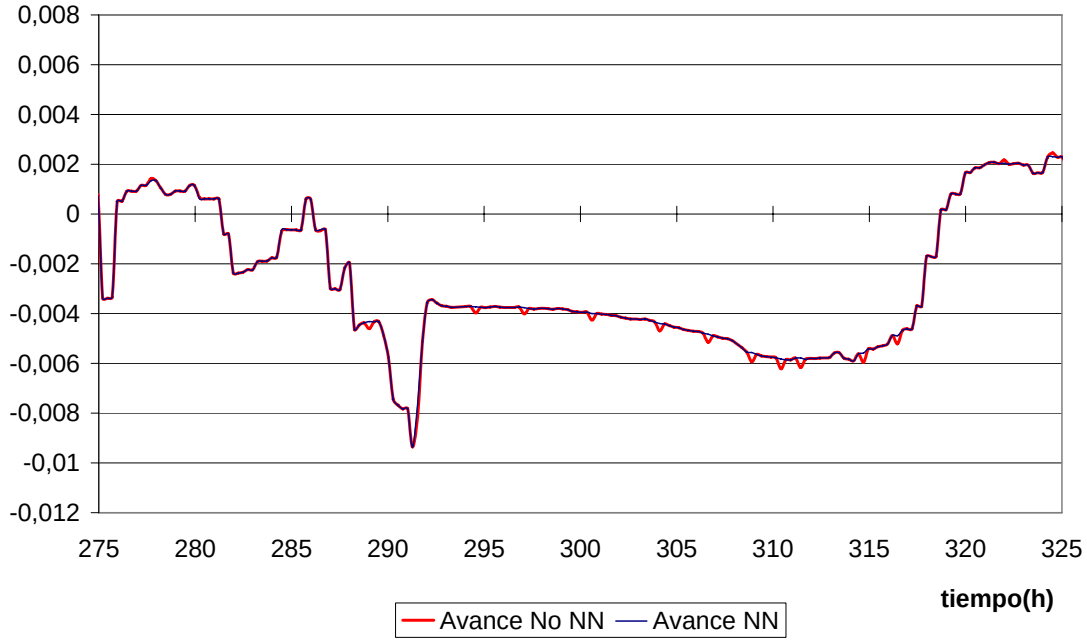

Figura 4.16 Comparación de la acción de los controladores.

Se aprecia que en varios puntos el controlador original reacciona con más brusquedad, pero como se señalaba anteriormente, al operar el sistema en relación a la acción de control derivada de la predicción no es posible comparar los resultados de los controladores.

Se ha obtenido un controlador anticipativo que emplea redes neuronales en su funcionamiento, que presenta un comportamiento estable. En muchos casos el control anticipativo tiene el inconveniente de que presenta un problema de variabilidad. Ese efecto no se ha notado en este caso, pero para ilustrar ese hecho se presenta a continuación un ejemplo similar basado en series de tiempo.

4.3.2.6. Comparación con Series de Tiempo

Se ha construido un modelo de serie de tiempo para la predicción de los datos de hidrógeno y caudal de metano. Los modelos resultante han sido AR(8) para el hidrógeno y AR(6) para el caudal de metano.

$$\mathrm{H2}_t = a_0 + \sum_{j=1}^{8} a_i \mathrm{H2}_{t-i} + \varepsilon_t^{\mathrm{H2}} \qquad (4.\ 14)$$

$$Q\mathrm{CH4}_t = b_0 + \sum_{j=1}^{6} b_i Q\mathrm{CH4}_{t-i} + \varepsilon_t^{\mathrm{QCH4}} \qquad (4.\ 15)$$

Se han implementado las predicciones en la estructura del controlador del mismo modo que anteriormente se emplearon las predicciones con redes.

Se comprobaron sus resultados en lazo cerrado. El perfil del experimento se muestra en la Figura 4.17

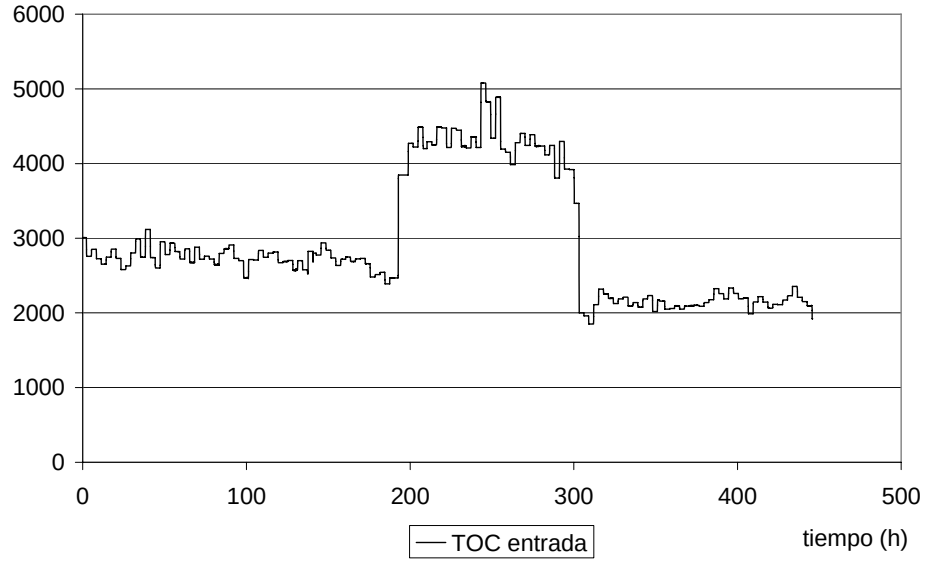

Figura 4.17 Sobrecargas del experimento de validación

Se llevaron a cabo una sobrecarga, y vuelta a la normalidad. La consigna objetivo era de 45 ppm de hidrógeno.

Una sección del comportamiento de las predicciones de hidrógeno y caudal de metano durante el experimento se muestran en las Figuras 4.18 y 4.19, respectivamente.

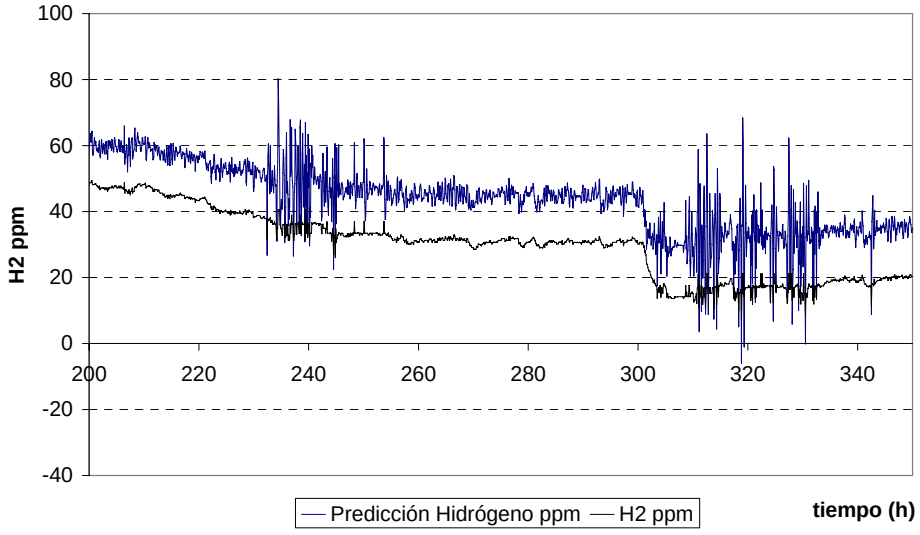

Figura 4.18 Sección del conjunto de validación. Hidrógeno.

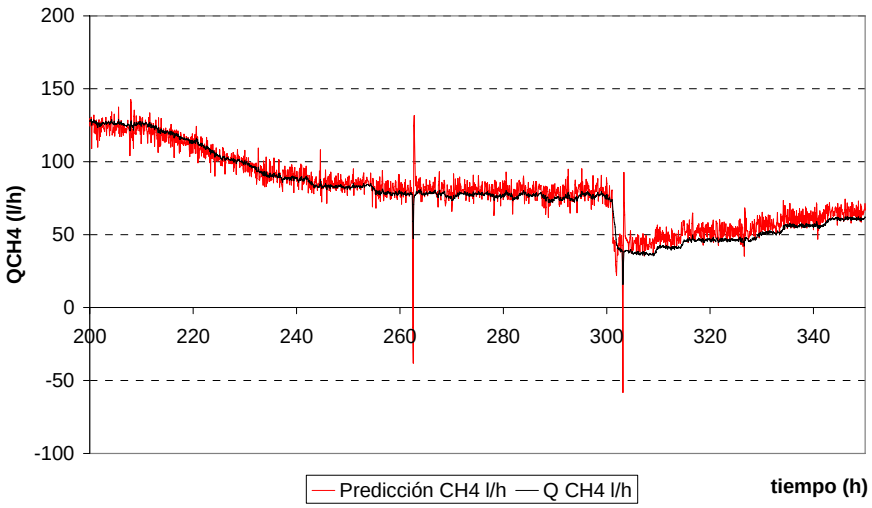

Figura 4.19 Sección del conjunto de validación. Caudal de metano.

De nuevo durante el funcionamiento del control el objetivo no es que las predicciones sean correctas sino hacer los ajustes necesarios para que no se cumpla lo previsto, por lo que no hay que juzgar las bondad de la estimación. En cualquier caso no se puede pasar por alto el sesgo que se aprecia en el comportamiento del hidrógeno. La causa de este desfase se debe a que un reactor anaerobio es un sistema que depende de las bacterias que trabajan en él; las sobrecargas, subcargas y las diferencias en la composición del sustrato pueden hacer que este ecosistema de organismos evolucione, y por lo tanto también lo haga el comportamiento de las variables que se miden en él. Es necesario por lo tanto una reestimación de los modelos frecuente. Otro de los fenómenos que se observan es la variabilidad de las predicciones, claramente mayor que en el caso de la predicción con redes neuronales.

El comportamiento del controlador fue el siguiente.

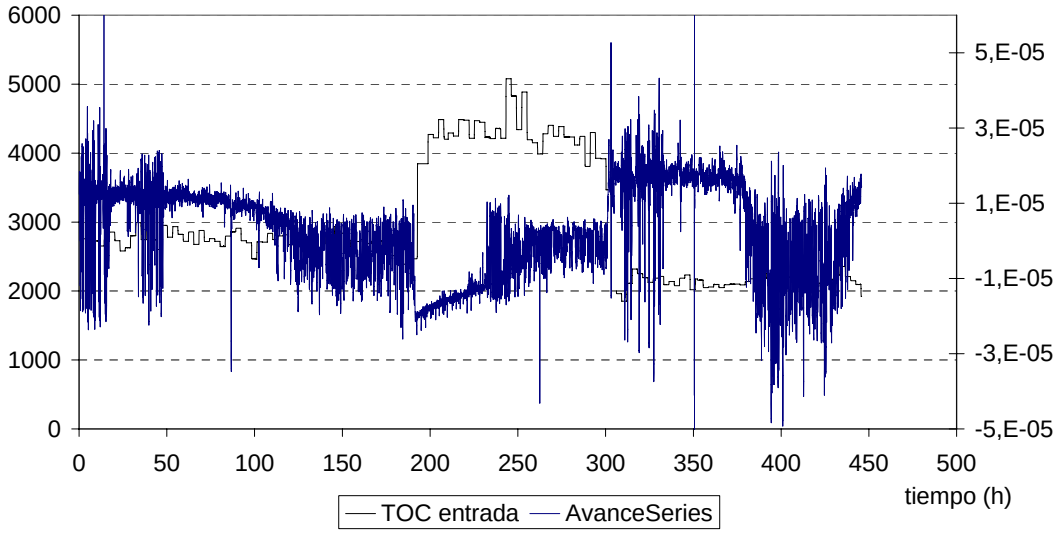

Figura 4.20 Acción de corrección del controlador.

Comparémoslo, como antes con el original, sin acción anticipativa. Se muestra la comparativa en una sección del conjunto de validación en la figura 4.21.

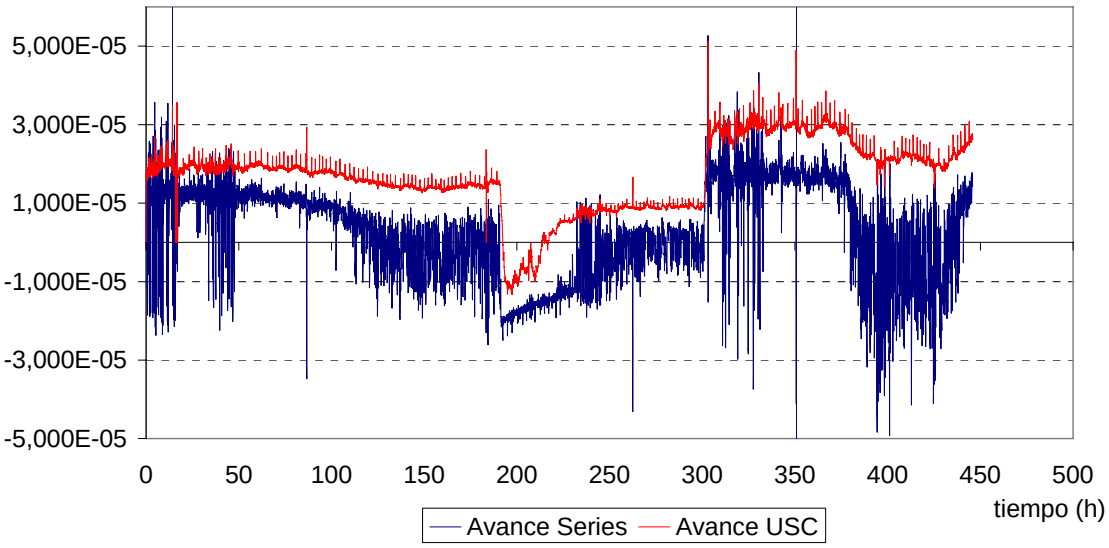

Figura 4.21 Comparación de la acción de los controladores.

En este caso, surge un problema, debido a que la consigna del experimento era de 45 ppm, un valor cercano a la predicción del hidrógeno, pero no a su valor real. Es por ello que las acciones del controlador son muy leves. Este es uno de los problemas del uso del control anticipativo, cuando, por cualquier circunstancia el comportamiento del sistema cambia y se produce un sesgo en la predicción. Se busca que la predicción esté cerca de la consigna, por lo que el valor real estará separado de su objetivo en ese sesgo de predicción.

Una de las limitaciones de trabajar con procesos reales, en particular cuando son, como este, delicados, es que en la práctica no es posible repetir dos veces las mismas situaciones. Es por ello que la comparación exacta entre ambos controladores no es posible con datos reales trabajando en online. Lo que sí se ha observado es que el controlador con redes neuronales parece ser menos inestable que el controlador con series de tiempo, debido, tal vez al modo de realizar la predicción multirretardo; como una curva completa en el caso neuronal, y como las sucesión e predicciones en el caso de series de tiempo.

Se ha evidenciado además la necesidad de reajuste de los modelos bien por reestimación de los parámetros o por bien por reentrenamiento, debido a la tendencia del sistema a evolucionar.

4.4. Conclusiones.

Se ha realizado un repaso de las técnicas de control más empleadas y se han presentado ideas de cómo introducir redes neuronales dentro de esas técnicas de control ya existentes. Se ha desarrollado ejemplos con datos reales aplicados a la industria y al medio ambiente, centrándose principalmente en el poder predictor de las redes neuronales.

En ambos enfoques se han obtenido resultados satisfactorios que muestran como las redes pueden ayudar en el control de procesos y en la detección de situaciones anómalas. En este campo existen todavía muchas líneas de trabajo por desarrollar, tanto en lo referente a la elaboración de modelos como al desarrollo de nuevos métodos de control que complementen a los ya existentes.

4.5. Bibliografía

Aynsley, M., Holfland, A., Morris, A.J., Montague G.A. y Di Massimo, C. (1993) Artificial Intelligence and the supervison of bioprocesses. Advances in Biotechnology, V.48(1), pp. 1-28.

Barto, A.G.(1900) Connectionist learning for control. Neural Networks for Control, chapter 1, pp. 5-58. MIT Press, Cambridge.

Bhat, N., y McAvoy, T.J. (1990). Use of neural nets for dynamic modeling and control of chemical process systems. Computer Chem Engng, V.14(4-5), pp.573-583.

Castellano, M., Ruiz, G., Gonzalez, W., Roca, E. & Lema, J. M. (2007) Selection of variables using factorial discriminant analysis for the state identification of an anaerobic UASB-UAF hybrid pilot plant, fed with winery effluents. Water Sci. Technol. V.56(10), pp. 139–145.

Chen,L., Bernard, O. Bastin, G. Angelov, P. (2000). Hybrid modelling of biotechnological processes using neural networks Control Engineering Practice, V.8, pp. 821-827.

Fernández, J.M. (1994) Tratamiento de aguas residuales de las industrias de tablero de fibra. Tesis doctoral Universidade de Santiago de Compostela. Santiago de Compostela, España.

Fernández, J.M., Méndez, R. y Lema J.M. (1995) Anaerobic treatment of Eucalyptus fibreboard manufacturing wastewater by a hybrid USBF lab-scale reactor. Environmental Technology, V.16(7), pp. 677-684.

Guwy, A.J., Hawkes, F.R., Wilcox, S.J. y Hawkes, D.L. (1997) Neural network and on-off control of bicarbonate alkalinity in a fluidised-bed anaerobic digester. Water Research, V.31(8), pp. 2019-2025

Hosogi, S. (1990) Manipulator control using layered neural network model with self-organizing mechanism. In International joint Conference on Neural Networks , V.2, pp. 217-220. Washington, DC. Lawrence Erlbaum.

Hulshoff Pol,L.W., Euler, H., Eitner, A. y Grohganz, D. (1997) GTZ sectoral project "Promotion of anaerobic technology for the treatment of municipal and industrial sewage and waste". In: Proc. 8th Int. Conf. on anaerobic digestion, V.2, pp. 285-292. Sendai, Japan.

Hunt, K.J., Sbarbaro, D., Zbikowski, R., Gawthrop P.J. (1992) Neural Networks for Control Systems -A Survey, Automatica, V.28(6), pp. 1083-1112.

Ichikawa, Y. y Sawa, T. (1992) Neural network application for direct feedback controllers. IEEE Transactions on Neural Networks, V.3(2), pp.224-231

Konstantinov, K.B. y Yoshida, T. (1992) Knowledge-based control of fermentation processes. Biotechnol. Bioeng., V.39, pp. 479–486

Lim, H.C. y Lee, K.S. (1991) Control of Bioreactor Systems. Biotechnology, Measuring, Modelling and Control. VCH, V. 4 . Schügerl, K., Editor.

Molina, F., Castellano, M., García, C., Roca, E. y Lema, J. M.(2009) Selection of variables for on-line monitoring, diagnosis, and control of anaerobic digestion processes. Water Science and Technology, V. 60(3), pp. 615-622.

Nahas, E.P., Henson, M.A. y Seborg, D.E. (1992), Non-linear Internal Model Control Strategy for Neural Network Models, Computers Chem. Engng., V.16(12), pp. 1039-1057.

Narendra, K.S. y Parthasaranthy, K. (1990) Identification and control of dynamic systems using neural networks, IEEE Trans. Neural Networks, V. 1(1), pp. 4-27.

Olsson, G. y Newell, B. (1999) Wastewater Treatment Systems. Modelling, Diagnosis and Control. IWA Publishing, London.

Psaltis, D, Sideris, A, Yamamura, AA. (1988) A multilayered neural network controller. IEEE Control Systems, V.8, pp. 17-21.

Psichogios D.C. y Ungar L.H. (1991) Direct and Indirect Model Based Control Using Artificial Neural Networks, Ind.Chem.Eng.Res, V.30, pp. 2564.2573.

Reyero, R. y Nicolás, C.F. (1995) Sistemas de control basados en lógica difusa:"Fuzzy control". Omron Electronics S.A. Centro de Investigaciones Tecnológicas IKERLAN, Madrid, España.

Rodríguez, J., Ruiz, G., Molina, F., Roca, E. y Lema, J.M. (2006) A hydrogen-based variable-gain controller for anaerobic digestion processes Water Science & Technology. V.54(2), pp. 57–62.

Ruiz G., Roca E. y Lema J.M. (2002) Selección de variables para la identificación de estados no estacionarios en la operación de reactores anaerobios. Apuntes del VII Taller y Simposio Latinoamericano sobre Digestión Anaerobia. Mérida– México. V.2, pp. 166-172.

Ruiz, G., Castellano, M., González, W., Roca, E. y Lema, J.M. (2005a) Transient state detection and prediction of organic overload in anaerobic digestion process using statistical tools.Computer Applications in Biotechnology 2004, pp. 357-362. M-N Pons and J van Impe (Eds). Elsevier. London.

Ruiz,G., Castellano,M., González, W., Roca, E. y Lema, J.M.(2005b) Anaerobic digestion process parameter identification and marginal confidence intervals by multivariate steady state analysis and bootstrap. In proceedings of: European Symposium on Computer Aided Process Engineering 15 (ESCAPE 15).

Shinskey, F.G. (1994) Feedback Controllers for the Process Industries, McGraw Hill,

Smith, C.A. y Corripio, A.B. (1991) Control Automático de Procesos. Teoría y Práctica, Limusa, México.

Stephanopoulos, G. y Han, C. (1996) Intelligent Systems in Process Engineering: a Review, Comp. Chem. Eng., V.20,(6/7), pp. 743-791.

Steyer,J.P., Pelayo-Ortiz,C., Gonzalez-Alvarez,V., Bonnet,B. y Bories,A. (2000) Neural network modelling of a depollution process. Bioprocess Engineering, V.23(6), pp. 727-730.

Switzenbaum, M.S. (1995) Obstacles in the implementation of anaerobic treatment technology. Bioresource Technology, v.53(3), pp. 255-262.

Werbos, P.J. (1990) Backpropagation through time: what it does and how to do it. Proc. IEEE, V.78(10), pp. 1550-1560.

Wilcox, S.J., Hawkes, D.L., Hawkes, F.R. y Guwy, A.J. (1995) A Neural- Network, Based on Bicarbonate Monitoring, to Control Anaerobic-Digestion. Water Research, V.29(6), pp. 1465-1470.

Young, J.C. y McCarty, L. (1969) The anaerobic filter for waste treatment. Journal of the Water Pollution Control Federation, V. 61, pp. 160-173.

Ziegler, J.G. y Nichols, N.B. (1942) Optimum Settings for Automatic Controllers. ASME Transactions, 64.