Regularización de redes neuronales tipo cascada correlación para el pronóstico de series de tiempo

Fernán Alonso Villa Garzón

Universidad Nacional de Colombia

Facultad de Minas, Departamento de Ciencias de la Computación y de la Decisión

Medellín, Colombia

2016

Regularización de redes neuronales tipo cascada-correlación para el pronóstico de series de tiempo

Fernán Alonso Villa Garzón

Tesis presentada como requisito parcial para optar al título de:

Doctorado en Ingeniería - Sistemas e Informática

Director:

Profesor Juan David Velásquez Henao, Ph.D

Línea de Investigación:

Investigación de Operaciones

Grupo de Investigación:

Grupo de Sistemas e Informática

Universidad Nacional de Colombia

Facultad de Minas, Departamento de Ciencias de la Computación y de la Decisión

Medellín, Colombia

2016

A mis padres Álvaro y Gloria, mis hermanos Alejandro y Camilo, mis Abuelos Manuel y Marina (Q.E.P.D), mi esposa Yulieth y a todos los familiares y amigos que en momentos olvidé por dedicarme a esto.

... necesité ingenio para proponer soluciones, paciencia para desarrollarlas, persistencia para probarlas, perseverancia para corregirlas, pasión para no abandonarlas, valor para continuar y sobre todo humildad para asumir éxitos y fracasos...

Agradecimientos

Deseo agradecer a las siguientes personas y entidades que fueron un apoyo importante para el desarrollo exitoso de esta tesis:

A mi director, mentor y amigo, Juan David Velásquez.

A todo mi entorno familiar.

A COLCIENCIAS, Universidad Nacional de Colombia – Facultad de Minas y Universidad de Antioquia – Facultad Nacional de Salud Pública.

A los jurados y revistas donde se presentaron algunos de los resultados de éste trabajo y a los jurados de tesis, con sus comentarios y revisión permitieron mejorar ostensiblemente este trabajo.

Finalmente, el desarrollo de un trabajo de doctorado deja varias lecciones espirituales y de vida, gracias a Dios, mediante la oración y la meditación pude salir adelante en momentos difíciles en los últimos cinco años de vida.

Resumen

En ésta tesis se ha diseñado un nuevo criterio para la selección del modelo de red neuronal Cascada-Correlación que controla íntegramente el tamaño del conjunto de entrenamiento, la complejidad del modelo y la magnitud de los pesos para obtener modelos con buena capacidad de generalización. Para esto se parte del análisis de los problemas que afectan la capacidad de generalización de la red: el primero relacionado con que la cantidad de datos del conjunto de entrenamiento contenga la información suficiente para reproducir el proceso generador de datos, el segundo con la cantidad de parámetros adecuada para representar el proceso generador de datos, y tercero la alta fluctuación de los parámetros; estos aspectos inciden en que el entrenamiento de la red sea un problema mal condicionado y se incurra en el sobreajuste. Para solucionar este tipo de problemas Tikhonov en (1963) propuso una solución matemática llamada Regularización; sin embargo, aplicar este tipo de solución es complejo dado que requiere determinar el parámetro de cuánto penalizar y el término de cómo penalizar y las propuestas que se han presentado han mostrado ser poco efectivas. Entonces, se ha propuesto desde un punto de vista conceptual y experimental cómo seleccionar efectivamente el valor del parámetro de regularización y dos nuevas ecuaciones de regularización con base en la varianza y la desviación estándar de los parámetros. Experimentalmente se mostró, al pronosticar siete series de tiempo, que la propuesta realizada es superior que modelos tradicionales de la literatura y que permite controlar efectivamente los problemas comentados y seleccionar el modelo adecuado que represente efectivamente a la serie de tiempo.

Palabras clave: Redes Neuronales Artificiales, Sobreajuste, Sobreparametrización, Criterio de Información, Redes Cascada-Correlación, Aprendizaje Predictivo, Series de Tiempo.

Abstract

In this dissertation, we have designed a new criterion for the selection of the cascade-correlation neural network model that controls integrally the size of the training set, model complexity and magnitude of the weights in order to obtain models with good generalization ability. We start with analysis of the problems affecting the generalization ability of the network, the first related to the amount of data in the training set that contains enough information to reproduce the data generating process, the second with the right amount of parameters to represent the data generating process and the third the high fluctuation of parameters. These three aspects affect the training of the network and make it an ill-conditioned problem and cause overfitting. To avoid these problems Tikhonov in (1963) proposed a mathematical solution called Regularization. However, implementing this type of solution is complex, since it requires the regularization parameter (how much to penalize) and determining the regularization term (how to penalize), the many proposals that have been presented them have proved ineffective. Then, we proposed from a conceptual and experimental view how to effectively choose the regularization parameter and as regularization terms the variance and standard deviation of the parameters. We show experimentally with seven time series, that our information criterion allows to effectively control the overfitting and allow to select the appropriate model that effectively represent the time series.

Keywords: Artificial Neural Networks, Overfitting, Over-parameterization, Information Criterion, Cascade-Correlation Networks, Predictive Learning, Time Series

Contenido

Lista de Figuras

Pág.

Lista de Tablas

Pág.

Introducción

El modelado y pronóstico de series de tiempo es un problema relevante desde el contexto académico hasta la industria. Básicamente, dado un conjunto de datos se requiere encontrar un modelo que permita capturar su comportamiento, con el fin de poderlo reproducir a futuro. Para lograr esto existen varias metodologías, marcos de trabajo y técnicas disponibles en la literatura las cuales dependerán de las características de los datos; por ejemplo si la supone que la media de los datos es lineal y su varianza constante, se recomienda usar modelos ARIMA; si la media es no lineal y varianza constante, modelos como redes neuronales artificiales. Una revisión de estos aspectos se encuentra en (Box & Jenkins, 1976; Montgomery, Johnson, & Gardiner, 1990; Wei, 2006).

El aprendizaje predictivo es un marco de trabajo general que permite encontrar modelos con capacidad predictiva, sus detalles se describen en el Capítulo 1; en este marco se encuentran las Redes Neuronales Artificiales y específicamente las arquitecturas de red Perceptrón Multicapa y Redes Cascada-Correlación; sin embargo, usar este marco de trabajo es complejo, dado que hay problemas metodológicos que deben resolverse antes y durante el entrenamiento o ajuste de sus modelos: el primero está relacionado con la cantidad de datos del conjunto de entrenamiento que contenga la información suficiente para reproducir el proceso generador de datos; el segundo con la cantidad de parámetros adecuada para representar el proceso generador de datos; y el tercero se relaciona con la alta fluctuación de los valores de los parámetros; estos aspectos inciden en que el entrenamiento de la red sea un problema mal condicionado y se incurra en el sobreajuste; para solucionar este tipo de problemas Tikhonov en (1963) propuso una solución matemática llamada regularización.

Aunque teóricamente la regularización es una solución apropiada, en la práctica muestra ser poco efectiva dado que para aplicar este tipo de solución se requiere determinar el

parámetro de cuánto penalizar (parámetro de regularización) y el término de cómo penalizar (término de regularización o penalización compleja) y las propuestas que se han presentado han mostrado ser poco efectivas, esta revisión se realiza en el Capítulo 1, mientras que en el Capítulo 2 se evidencia experimentalmente el problema del sobreajuste y las complicaciones para controlarlo.

Tradicionalmente se ha indicado que el término o estrategia de regularización de descomposición de pesos es efectivo para controlar el sobreajuste; sin embargo, se muestra, con evidencia experimental, que no controla adecuadamente la varianza de los parámetros y por ende no controla adecuadamente el sobreajuste, lo cual es un requisito primordial para que se pueda contemplar la regularización como un criterio de información.

Con el fin de controlar efectivamente el problema del sobreajuste y seleccionar adecuadamente el modelo para una serie en particular, en esta Tesis se ha propuesto desde un punto de vista conceptual y experimental cómo seleccionar efectivamente el parámetro de regularización y como términos de regularización la varianza y la desviación estándar de los parámetros, lo cual conforma el nuevo criterio de información para redes neuronales tipo cascada-correlación, el cual es descrito en el Capítulo 4.

Mediante la evaluación experimental, detallada en el Capítulo 3 y 4, pronosticando seis series de tiempo tradicionales usadas tanto por Makridrakis *et al.* (1998) como por Ghiass *et al.* (2005), se demostró que la capacidad de generalización del criterio de información propuesto es superior a la de modelos tradicionales de la literatura (DAN2, MLP, ARIMA) y que permite controlar efectivamente los problemas que generan el sobreajuste y seleccionar el modelo adecuado que represente efectivamente a la serie de tiempo.

Desde un punto de vista académico, los resultados de la investigación tienen un impacto asociado al desarrollo de nuevos conocimientos tanto en el área de la Inteligencia Computacional como en el área de la Econometría. Su impacto en el área de Inteligencia Computacional está relacionado con el desarrollo de un nuevo criterio de información basado en la teoría y la práctica. La Econometría se ve impactada, ya que es un trabajo que permite consolidar aún más el uso de las redes Cascada-Correlación al conjunto de técnicas usada para el modelado y la predicción de series temporales, así como por la

aplicabilidad práctica que estas tienen. Ello es debido a que hay un valor agregado relacionado con la utilidad del modelo para representar la dinámica de series temporales, y las aplicaciones posteriores de estos, como por ejemplo, en la valoración de políticas.

Los resultados también tienen un impacto importante al interior del grupo de investigación de Sistemas e Informática, ya que fortalece el área de inteligencia computacional y genera una experiencia muy importante en el modelado de series temporales. Igualmente, fortalece el grupo de investigación al darle continuidad a su trabajo, y da pie a la formulación de nuevos proyectos en el mediano plazo.

1. Capítulo 1: Sobre el aprendizaje

En éste Capítulo se revisan diversos aspectos teóricos y prácticos del aprendizaje predictivo con redes neuronales artificiales (RNA), específicamente para el pronóstico de series de tiempo con perceptrones multicapa y redes cascada-correlación, que son relevantes para el desarrollo de esta Tesis. Además, se presentan las principales limitaciones que se deben enfrentar al usar este tipo de modelos, especialmente el problema del sobreajuste, para el cual se han consultado cuáles han sido los avances para su solución y se analiza por qué tales soluciones no son efectivas; con base en lo anterior, surge el objetivo principal de esta tesis: proponer una solución efectiva al problema del sobreajuste para redes neuronales tipo Cascada-Correlación.

1.1 El aprendizaje predictivo

En campos como la econometría, las finanzas, la estadística, la sismología, la meteorología, la epidemiología se requiere encontrar modelos que describan sus procesos o fenómenos con el fin de realizar pronósticos o simulaciones que permitirán a sus agentes tomar decisiones objetivas y asertivas en el corto, mediano y largo plazo; además, estos modelos permiten realizar la gestión de la información, lo que facilita la gestión y descubrimiento de conocimiento y la toma de decisiones y generación de políticas asertivas (Cherkassky, 2012; Awad & Ghaziri, 2008; Palit & Popovic, 2005; Duda, Hart, & Stork, 2000; Cios & Pedrycz, 1998)

Los datos generados por algún proceso o fenómeno (el cual se seguirá llamando Proceso Generador de Datos, PGD) son el insumo fundamental para encontrar un modelo; específicamente, para construir un modelo con fines de pronóstico, el tipo de colección de datos de interés son las series de tiempo, su pronóstico es considerado un problema común en muchas disciplinas del conocimiento y su interés aumentado en la última década, especialmente debido a tendencias como Big Data (Montgomery & Jenni, 2015; Larose & Larose, 2015; Palit & Popovic, 2005). Un ejemplo tradicional: en la

administración de la producción y sistemas de inventario se realizan frecuentemente pronósticos con el fin de facilitar la toma de decisiones de corto, mediano y largo plazo sobre procesos de control de calidad, análisis de inversiones, planeación financiera, mercadeo, entre otros (Montgomery, Johnson, & Gardiner, 1990).

El aprendizaje predictivo (también conocido en la literatura como aprendizaje con base en observaciones, aprendizaje inductivo o aprendizaje dependiente de los datos) permite encontrar modelos con fines de pronóstico y ha sido estudiado tradicionalmente en diversos campos (ver Tabla 1-1); Friedman en 1994 propuso una perspectiva común que unificó los principios bajo los cuales se habían desarrollado los métodos presentados en la Tabla 1-1, este fue uno de los primeros aportes para consolidar y unificar el conocimiento generado hasta ese momento (Friedman, 1994). Más tarde en el 2007, con el aporte y diversidad de nuevos desarrollos, Cherkassky y Mulier establecieron un marco conceptual general para este tipo de aprendizaje; uno de los aportes considerablemente importantes de este marco, es la recapitulación y consolidación de los principios fundamentales sobre los cuales se deben basar nuevos métodos que se propongan para el aprendizaje en campos como la estadística y la ingeniería (Cherkassky & Mulier, 2007). Tempranamente, Cherkassky en el 2012, reliza un nuevo análisis donde contrasta el aprendizaje predictivo con tendencias como el descubrimiento de conocimiento y la filosofía de las ciencias, y presenta la interpretación de modelos basados en el análisis de datos bajo el marco de trabajo de aprendizaje predictivo (Cherkassky, 2012).

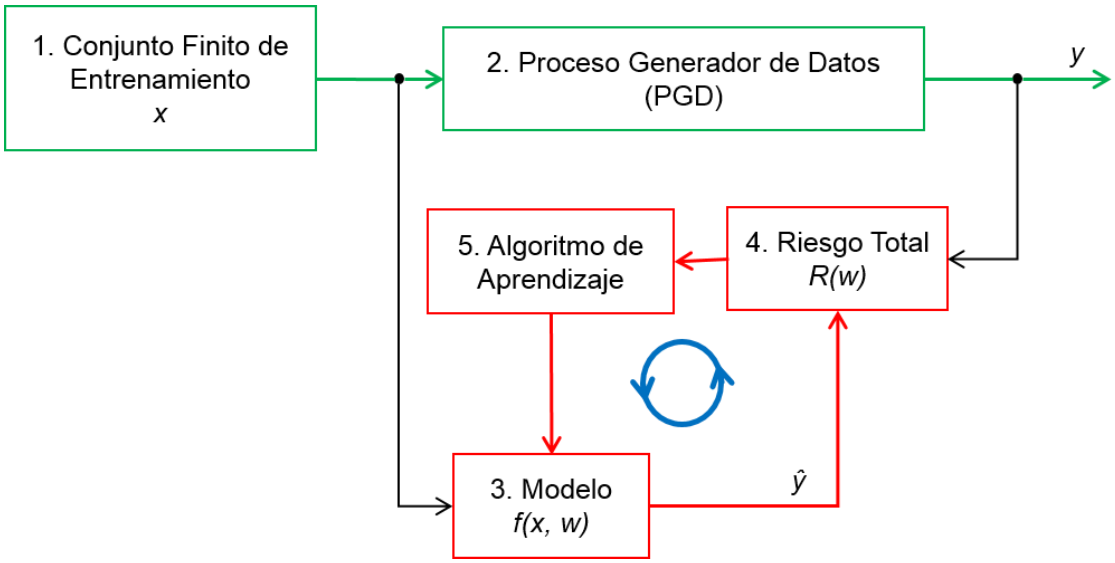
Tabla 1-1: Campos y correspondientes métodos para el aprendizaje predictivo. (Friedman, 1994)

Campo	Método
Matemáticas Aplicadas	Aproximación de Funciones
Estadística	Regresión no paramétrica
Ingeniería	Reconocimiento de Patrones
Inteligencia Artificial	Aprendizaje de Máquinas
Conexionismo	Redes Neuronales

Entonces, con base en los trabajos de Friedman (1994), Cherkassky *et al.* (2007) y Cherkassky *et al.* (2012), el aprendizaje predictivo se especifica como un proceso cuyo principal objetivo es encontrar un modelo que imite apropiadamente un fenómeno

determinado (Cherkassky & Ma, 2006) y que, además, tenga la capacidad de tener "buena" generalización de datos futuros (Cherkassky & Mulier, 2007; Cherkassky, 2012). Éste proceso está compuesto por dos fases, la primera corresponde a la entrenamiento (ver Figura 1-1) y la segunda a la de evaluación, también conocida como validación (ver Figura 1-2).

Figura 1-1: Entrenamiento, primera fase del aprendizaje predictivo.

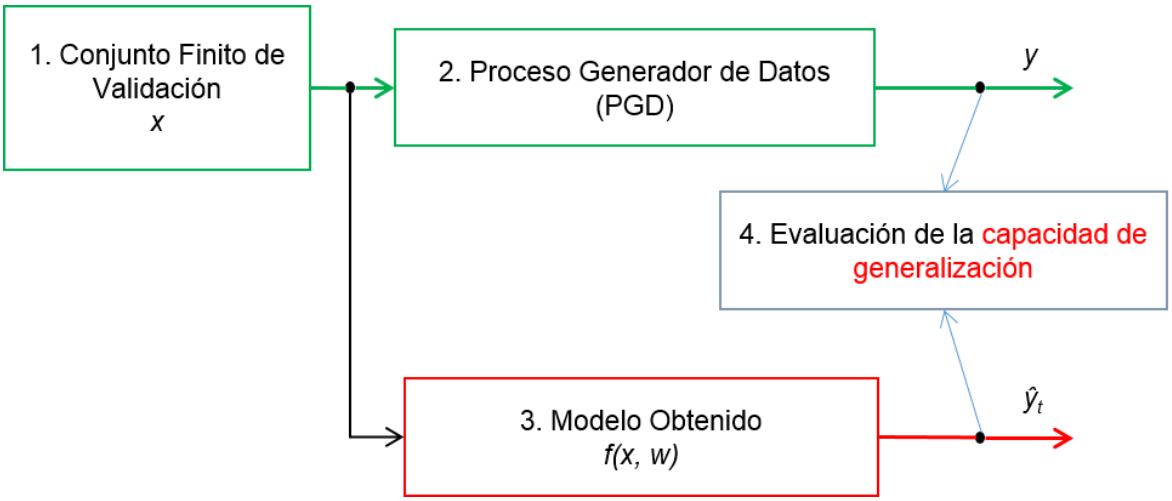


El objetivo de la primera fase, representada en la Figura 1-1, es encontrar un modelo matemático mediante una función *f* que represente el comportamiento de un proceso generador de datos; en esta fase se consideran cinco componentes indispensables:

1. Un conjunto finito de datos de entrenamiento *x*, los cuales corresponden a los datos de entrada.
2. Un proceso generador de datos (*PGD*), el cual corresponde al fenómeno que genera una salida real *y* a partir del conjunto *x*.
3. Un modelo procedente de una función *f(x, w)*, seleccionada *a priori*, con la cual se procurará representar y reproducir el comportamiento del PGD, esta genera una salida estimada *ŷ* a partir del conjunto *x* y parámetros *w*. Adicionalmente, $w \epsilon \Omega$, donde Ω es un conjunto de parámetros abstractos (Cherkassky & Mulier, 2007).

4. Una función de medida del Riesgo Total *R(w)*, seleccionada *a priori*, la cual permitirá evaluar la diferencia entre la salida real *y* la salida *ŷ*, es decir, qué tan cercana es la salida estimada a la salida real.
5. Un algoritmo de aprendizaje, seleccionado *a priori,* el cual permitirá estimar los parámetros *w* de la función *f*, minimizando *R(w),* tal que la salida estimada *ŷ* sea lo más cercana posible a la salida real *y*. En sentido común, este algoritmo tiene la responsabilidad de encontrar sistemáticamente cuáles son los parámetros que mejor se ajustan en *f* para obtener la mejor *ŷ* posible.

Figura 1-2: Evaluación, segunda fase del aprendizaje predictivo



Una vez se ha encontrado un modelo *f(x,w)*, se requiere evaluar su capacidad de generalización de datos desconocidos, es decir, valorar qué tanta capacidad de pronóstico tiene el modelo hallado (Figura 1-2). Para esto se plantean cuatro componentes indispensables:

1. Un conjunto finito de datos de validación *x,* diferente al de la primera fase.
2. El modelo *f(x, w)* obtenido en la primera fase.
3. El proceso generador de datos (*PGD*) de la primera fase, que genera una salida real *y,* desconocida para *f(x, w),* a partir del conjunto *x* de validación.

4. Una función de medida de validación o de evaluación de la capacidad de generalización, seleccionada *a priori*, la cual permitirá evaluar la diferencia entre la salida real *y* y la salida $\hat{y}$, es decir, qué tan cercana es la salida estimada a la salida real.

Sin embargo, en el proceso de aprendizaje predictivo, fases uno y dos, aunque se garantice que el algoritmo de aprendizaje encuentre los valores de los parámetros para obtener el óptimo global, esto no es suficiente para encontrar modelos con adecuada capacidad de generalización para un conjunto de datos específicos, dado que este proceso se encuentra influenciado por tres factores:

1. El tamaño y calidad del conjunto de entrenamiento (*x*). Se requiere que éste contenga la información suficiente para modelar y reproducir el PDG. Se pueden presentar varias situaciones al seleccionar este conjunto; entre estas, si es relativamente pequeño es posible que no contenga la información suficiente para modelar y reproducir el PDG; si por el contrario, es relativamente grande es posible que contenga más información de la necesaria (ruido, redundancia, puntos atípicos, etc.) que afectaría negativamente el entrenamiento. Entonces, se requiere tener un conjunto de entrenamiento de tamaño considerable que contenga la mayor cantidad de información posible que represente la estructura del comportamiento del PDG (Jin Cui, Davis , Cheng, & Xue, 2004).
2. La complejidad del modelo. Se requiere determinar la cantidad adecuada de parámetros que permita representar el comportamiento del PDG, es decir, la selección de la topología del modelo incide directamente en la capacidad de generalización del modelo (Cherkassky, 2012; Villa, Velásquez, & Souza, 2008; Flórez López & Fernández, 2008; Haykin, 1999; Hush & Horne, 1993).
3. La complejidad propia del PGD. Los datos generados por el proceso pueden contener el efecto de variables exógenas (*i.e.* datos extremos o atípicos) lo cual eleva la varianza de los datos y degrada la capacidad de generalización del modelo; otros aspectos que se deben considerar son: el desconocimiento del proceso y de sus componentes o variables; la incertidumbre, no linealidad y ruido en los datos generados (Cherkassky & Mulier, 2007; Suykens, Van Gestel, Brabanter, Moor, & Vandewalle, 2005; Bishop, 1994).

Entonces, con base en el marco de trabajo que se ha definido, se analizarán a continuación los aspectos de aprendizaje predictivo aplicados a las Redes Neuronales Artificiales para el pronóstico de series de tiempo.

1.2 El aprendizaje predictivo con redes neuronales para series de tiempo

1.2.1 Motivación

Una serie de tiempo se define como una secuencia de observaciones de un fenómeno determinado, ordenadas secuencialmente y registradas, usualmente, en igual intervalo de tiempo. El modelado de una serie consiste en construir sistemáticamente una representación matemática que permita capturar, total o parcialmente el comportamiento del PGD; una vez se construye y se estima un modelo, es posible realizar el pronóstico de la serie para un horizonte determinado, es decir, estimar sus valores futuros (Montgomery, Johnson, & Gardiner, 1990; Lachtermacher & Fuller, 1995; Bowerman, O'Connell, & Koehler, 2006).

El pronóstico de series de tiempo ha sido tratado con diferentes tipos de modelos estadísticos y matemáticos (Montgomery & Jenni, 2015; Wei, 2006; Palit & Popovic, 2005; Kasabov, 1998); los cuales, en un sentido amplio, se pueden categorizar en lineales y no lineales con base en un comportamiento supuesto para una serie de tiempo (Palit & Popovic, 2005). Para modelar series de tiempo con comportamiento supuesto como lineal se han usado ampliamente modelos como: AR, MA, ARMA y ARIMA (Box & Jenkins, 1976; Montgomery, Johnson, & Gardiner, 1990; Wei, 2006). Sin embargo, éste tipo de modelos no es suficiente, dado que la gran mayoría de series de tiempo en ingeniería, finanzas y econometría presentan un comportamiento aparentemente no lineal (Palit & Popovic, 2005).

En la literatura más relevante se han propuesto diversos modelos no lineales entre los que se encuentran: bilineales (Granger & Anderson, 1978); autorregresivos de umbral (TAR) (Tong & Lim, 1980); de heterocedasticidad condicional autorregresiva (ARCH) (Engle, 1982); autorregresivos de transición suave (STAR) (Chan & Tong, 1986; Dick van Dijk, Teräsvirta, & Franses, 2002; Tong, 2011); de heterocedasticidad condicional

autorregresiva generalizada (GARCH) (Bollerslev, 1986). Adicionalmente a esta breve revisión, Tong (1990), De Gooijer y Kumar (1992), Peña (1994), Tjostheim (1994), Hardle *et al.* (1997), Tong (2011) y Montgomery y Jenni (2015) realizan una amplia recopilación donde examinan otros modelos.

Aunque los modelos no lineales tradicionales han demostrado ser útiles en problemas particulares, no son adecuados para la mayoría de los casos, dado que suponen una forma de no linealidad preestablecida en la serie, es decir, los datos se deben adaptar a la estructura no lineal definida por el modelo; de este modo, muchas veces no representan adecuadamente el comportamiento de la serie, véase a (Granger & Teräsvirta, 1993). Además, para definir cada familia de estos modelos, es necesario especificar un tipo apropiado de no linealidad; esto es una tarea difícil comparado con la construcción de modelos lineales; la cantidad posible de funciones para definir el tipo de no linealidad es amplia (Granger, 1993; Zhang, Patuwo, & Hu, 2001).

Por otro lado, desde la Inteligencia Computacional, *i.e.* aprendizaje de máquinas, se han propuesto diversas técnicas para el modelado y pronóstico de series de tiempo; de las disponibles, las redes neuronales artificiales (RNA) han mostrado ser más robustas que otras técnicas tradicionales, especialmente en la representación de relaciones complejas que exhiben comportamientos no lineales, por ejemplo véase a Ghiassi *et al.* (2005) y Villa *et al.* (2008). Masters (1993), recomienda utilizar RNA en vez de alguna técnica tradicional por las siguientes razones:

- Poseen una amplia capacidad para aprender relaciones desconocidas a partir de un conjunto de ejemplos.
- Tienen una alta tolerancia a patrones extraños de ruido y componentes caóticas presentes en un conjunto de datos.
- Son suficientemente robustas para procesar información incompleta, inexacta o contaminada.
- No restringen el tipo de no linealidad de la serie de tiempo a la estructura matemática del modelo de red neuronal.

Respecto al pronóstico de series de tiempo con RNA, Zhang *et al.* (1998) realizaron una revisión general del estado del arte donde resaltan tanto éxitos y fracasos reportados de las redes neuronales (especialmente con los perceptrones multicapa); incluyendo las

publicaciones más relevantes y los tópicos de investigación más influyentes hasta 1996. Sin embargo, en la última década se ha producido un considerable número de contribuciones en múltiples campos como metodologías de aprendizaje, selección de entradas relevantes, neuronas ocultas, entre otros, cuya influencia no ha sido evaluada ni reportada en la literatura. Además, Cogollo y Velázquez (2014) presentan los avances metodológicos que se han realizado entre el 2000 y 2010 en el pronóstico de series de tiempo con redes neuronales artificiales.

Desde el análisis predictivo (también traducido como analítica predictiva) (Larose & Larose, 2015), tradicionalmente se han usado varios tipos de RNA para el pronóstico de series de tiempo, entre estos se encuentran: perceptrón multicapa (MLP) (Haykin, 1999; Palit & Popovic, 2005); funciones de base radial (RBF) (Zhang, Han, Ning, & Liu, 2008; Yan, Wang, Yu, & Li, 2005); máquinas de soporte vectorial (SVM) (Cao & Tay, 2003); *FNN - Fuzzy Neural Networks* (Rast, 1997; Kadogiannis & Lolis, 2002); *Feedforward and Recurrent Networks* (Gençay & Liu, 1997; Parlos, Rais, & Atiya, 2000; Mishra & Patra, 2009).

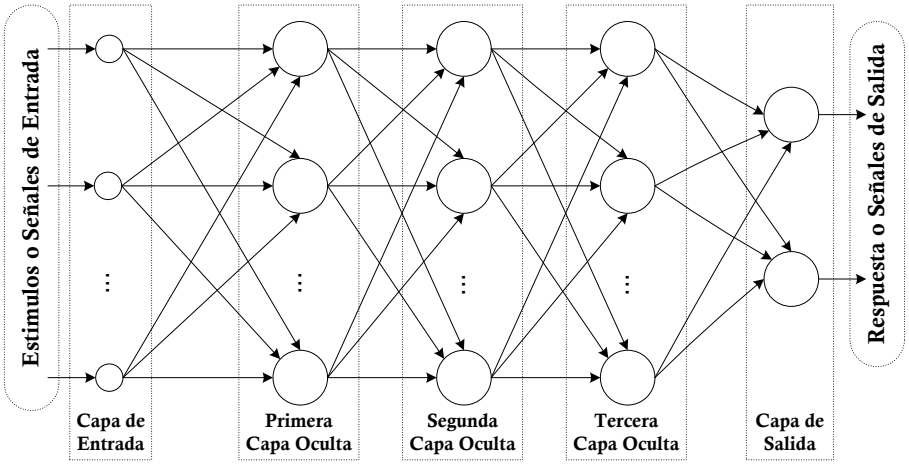
Zhang *et al.* (1998) y Palit *et al.* (2005) coinciden en que el tipo de red neuronal MLP es uno de los tipos más apropiados para el modelado y pronóstico de series de tiempo; actualmente, es usado en las nuevas tendencias de análisis de datos como *Big Data* (Larose & Larose, 2015). Su éxito se debe a que, desde un punto de vista matemático, un MLP tiene la capacidad de aproximar cualquier función continua definida en un dominio compacto con una precisión arbitraria previamente establecida (Hornik, Stinchcombe, & White, 1989; Cybenko, 1989; Funahashi, 1989); además, en la práctica, los MLP se han caracterizado por ser muy tolerantes a información incompleta, inexacta o contaminada con ruido, y por descubrir y aprender relaciones desconocidas (Masters, 1993).

1.2.2 Los Perceptrones Multicapa

Un MLP es un tipo de red neuronal que imita la estructura masivamente paralela de las neuronas del cerebro. Básicamente, es un conjunto de neuronas (nodos) que están lógicamente ordenadas en tres o más capas; generalmente, posee una capa de entrada, una oculta y una de salida, cada una de éstas tiene al menos una neurona. Entre la capa

de entrada y la capa de salida, es posible tener una o varias capas ocultas, como se muestra en la **Figura 1-3**; aunque se ha demostrado que para la mayoría de problemas es suficiente con una sola capa oculta (Palit & Popovic, 2005); mientras que para el pronóstico de series de tiempo es suficiente con una neurona en la capa de salida (Villa & Velásquez, 2010).

Figura 1-3: Representación gráfica de una arquitectura básica de un MLP con una capa de entrada, tres ocultas y una de salida.



Para pronosticar una serie de tiempo con un MLP, se toma como punto de partida, que una serie se define como una secuencia de T observaciones ordenadas en el tiempo:

$$y_t = \{y_i\}_1^T \tag{1.1}$$

Para la cual que se pretende estimar una función que permita explicar y_t en función de sus rezagos $\{y_{t-1}, y_{t-2},\ldots,y_{t-p}\}$; es posible especificar matemáticamente la función y_t como un MLP, así:

$$y_t = \beta_0 + \sum_{h=1}^{H} \beta_h \times g\left(\alpha_h + \sum_{p=1}^{P} w_{p,h} \times y_{t-p}\right) + \varepsilon_t \tag{1.2}$$

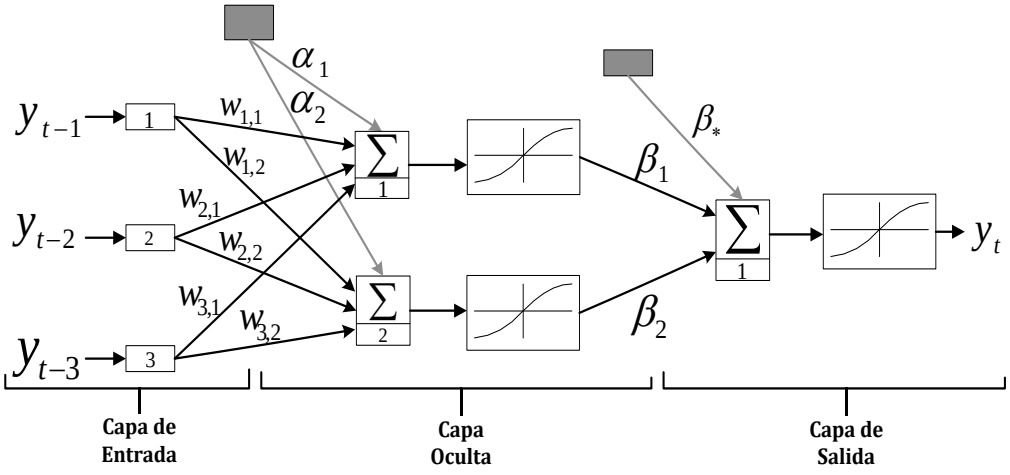
La Ecuación (1.2) equivale a un modelo estadístico no paramétrico de regresión no lineal (Sarle, 1994); para ésta ecuación se tienen en cuenta los siguientes aspectos: se asume que ε_t sigue una distribución normal con media cero y varianza desconocida σ^2; H

representa el número de neuronas en la capa oculta; *P* es el número máximo de rezagos considerados (neuronas de entrada); $g(\cdot)$ es la función de activación de las neuronas de la capa oculta.

Además, los parámetros $\Omega = [\beta_*, \beta_h, \alpha_h, w_{p,h}]$, con $h = 1, 2, \ldots, H$ y $p = 1, 2, \ldots, P$, son estimados usando el principio de máxima verosimilitud de los residuales, el cual equivale a la minimización del error cuadrático medio. En el contexto de las series de tiempo, el modelo puede ser entendido como una combinación lineal ponderada de la transformación no lineal de varios modelos autorregresivos.

Con base en la Figura 1-3 y la Ecuación (1.2), en la Figura 1-4 se exhibe una representación pictórica de un modelo con tres rezagos, dos neuronas en la capa oculta y una en la capa de salida; en ésta se puede tener la visión holística de la relación de dependencia entre y_t, sus rezagos y los parámetros que deben ser estimados.

Figura 1-4: Perceptrón Multicapa con tres neuronas en la capa de entrada, dos en la oculta y una en la salida.



Sin embargo, la especificación de un MLP es un proceso complejo dado que es necesario definir una serie de parámetros *a priori* con criterio experto, entre estos, la cantidad de neuronas en la capa de entrada (rezagos) *P*, la cantidad de neuronas de la capa oculta *H*, la función de activación $g(\cdot)$, el método para estimar los parámetros, la función objetivo a optimizar (*e.g.* SSE, MSE, RMSE, MAE, GRMSE), el método para evaluar que el modelo estimado representa adecuadamente el comportamiento la serie

de tiempo, entre otros (Flórez López & Fernández, 2008; Palit & Popovic, 2005; Kaastra & Boyd, 1996).

Sobre la estimación los parámetros Ω del modelo definido en la Ecuación (1.2), esta puede plantearse como problema numérico de optimización (Masters, 1993), mientras que desde un punto de vista estadístico, es un proceso de estimación no paramétrica funcional (Chow & Cho, 2007). Para resolverlo se han propuesto diversas técnicas de optimización: basadas en gradiente, tales como *Backpropagation* (Riedmiller & Braun, 1993), y *RPROP - Resilient Backpropagation* (Riedmiller & Braun, 1993; Riedmiller, 1994); heurísticas, como estrategias evolutivas (Ortíz, Villa, & Velásquez, 2007), entre otras.

En general, RPROP es considerado como uno de los algoritmos basados en gradiente más apropiados para entrenar redes neuronales artificiales (Ortíz, Villa, & Velásquez, 2007; Riedmiller & Braun, 1993; Riedmiller, 1994).

Sin embargo, el problema no es simplemente estimar cada modelo para una serie en particular. Mientras en el caso lineal hay una importante experiencia ganada, existen muchos problemas teóricos, metodológicos y empíricos abiertos sobre el uso de modelos no lineales. En el caso del MLP, su proceso de especificación es difícil debido a la complejidad de los pasos metodológicos que requiere:

1. Seleccionar cuáles son las entradas al modelo o rezagos (neuronas capa de entrada).
2. Determinar la cantidad de neuronas en la capa oculta.
3. Seleccionar la función de activación.
4. Seleccionar cuál es la función objetivo que se desea optimizar (*e.g.* SSE, MSE, RMSE, MAE, GRMSE).
5. Estimar los parámetros del modelo con alguna técnica de optimización.
6. Cómo evaluar la capacidad de generalización del modelo, es decir, validar que el modelo estimado representa adecuadamente el comportamiento de la serie.

A lo anterior, se suma la dificultad de que los criterios sobre cómo abordar cada paso son subjetivos (Kaastra & Boyd, 1996); y la falta de identificabilidad estadística del modelo es uno de los aspectos que dificultan su especificación. Esto se relaciona con que los

parámetros óptimos no son únicos para una especificación del modelo (número de entradas o rezagos, cantidad de neuronas en la capa oculta, funciones de activación, etc.), y un conjunto de datos dado. Esto se debe a que (Anders & Korn, 1999):

- Se puede obtener múltiples configuraciones que son idénticas en comportamiento cuando se permutan las neuronas de la capa oculta, manteniendo vinculadas las conexiones que llegan a dichas neuronas.
- Cuando las neuronas de la capa oculta tienen funciones de activación simétricas alrededor del origen, la contribución neta de la neurona a la salida de la red neuronal se mantiene igual si se cambian los signos de los pesos que entran y salen de dicha neurona.
- Si los pesos de las conexiones entrantes a una neurona oculta son cero, es imposible determinar el valor del peso de la conexión de dicha neurona oculta a la neurona de salida.
- Si el peso de la conexión de una neurona oculta hacia la neurona de salida es cero, es imposible identificar los valores de los pesos de las conexiones entrantes a dicha neurona oculta.

Otro inconveniente que se debe tener en cuenta, al igual que los modelos tradicionales, es que los MLP pueden adolecer del fenómeno del sobreajuste; cuando la red está sobreparametrizada, es decir, cuando ella tiene más neuronas de las necesarias para solucionar el problema, no generaliza adecuadamente los datos y sólo puede responder correctamente ante los estímulos ya conocidos; es decir, la red memoriza los datos de entrenamiento en vez de aprender; esto se evidencia cuando se produce un error de entrenamiento muy pequeño y un error de validación muy alto (Villa, Velásquez, & Souza, 2008).

En el contexto general del modelado y pronóstico de series de tiempo, se concluye que los MLP presentan serias limitaciones debido a que su proceso de especificación es difícil debido a la complejidad de los pasos metodológicos que requiere (selección de la entradas al modelo, cantidad de neuronas en la capa oculta, etc.) (Sánchez & Velásquez, 2010). Y al igual que los modelos tradicionales (no paramétricos y no lineales), los MLP pueden adolecer del fenómeno del sobreajuste, y memorizar los datos de entrada degradando su capacidad de pronóstico (Masters, 1993).

1.2.3 Las Redes Cascada-Correlación

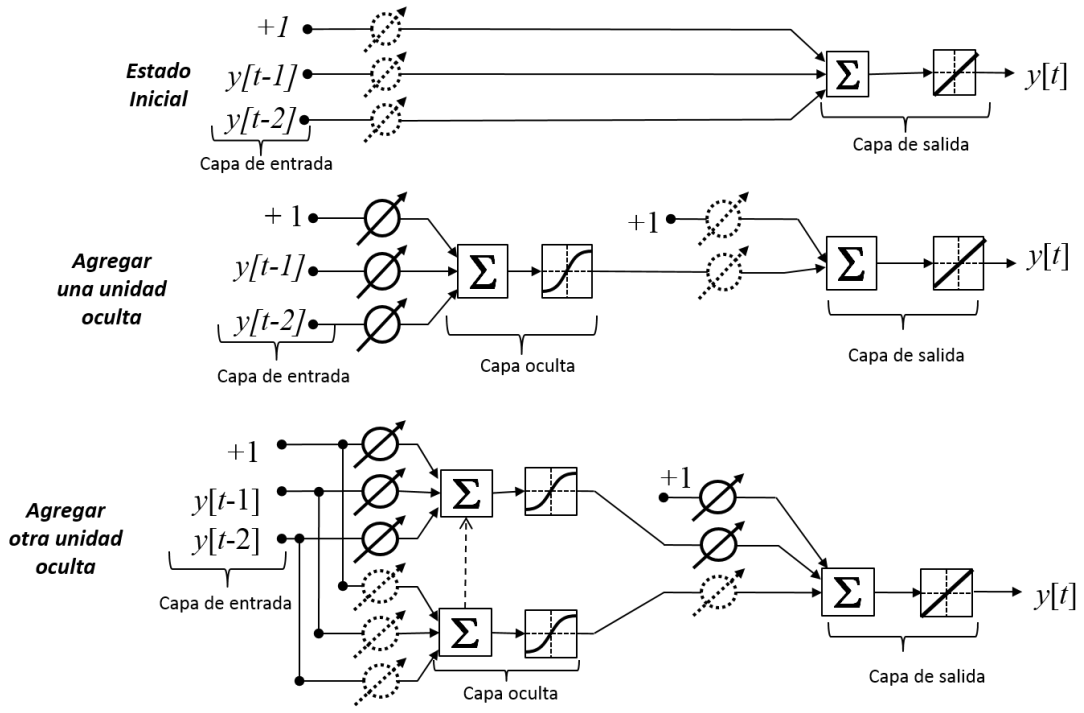
La red neuronal artificial conocida como Cascada-Correlación (CC) (Fahlman & Lebiere, 1990) presenta ventajas conceptuales en relación al proceso de especificación de los MLP. Combina dos ideas básicas: el aprendizaje incremental (también conocido como constructivo o de crecimiento de red), donde se agrega una neurona oculta a la vez, y no se modifican después de haberse agregado; y la arquitectura en cascada, donde para cada nueva neurona oculta, el algoritmo trata de maximizar la magnitud de la correlación entre la nueva neurona oculta y el error residual de la red.

Entonces, en las redes CC, se comienza con una red mínima sin capas ocultas, es decir, con sólo algunas entradas (predefinidas y fijas) y uno o más nodos de salida; las neuronas ocultas son agregadas una a una en la red, obteniendo de esta manera una estructura multicapa. En el proceso de adición de neuronas ocultas a la red, cada nueva neurona recibe una conexión sináptica de cada una de las neuronas de entrada y también de las neuronas ocultas que la preceden. Luego de agregar la nueva neurona oculta, los pesos sinápticos de su entrada son congelados, mientras que los pesos de su salida son entrenados repetidamente. Este proceso continúa hasta que se alcanza un rendimiento deseado. En la Figura 1-5 se presenta el esquema de una red CC, donde las conexiones con símbolos punteados indican los pesos (subconjunto de parámetros Ω de la Ecuación (1.2) que deben ser actualizados; mientras que las otras conexiones (*i.e.* símbolos sin puntear) indican los parámetros que son congelados una vez se ha agregado una unidad en la capa oculta.

Resulta evidente que la arquitectura en cascada permite agregar cada neurona oculta a la vez y sólo se actualizan o estiman los nuevos pesos; además, el aprendizaje incremental o constructivo permite crear e instalar las nuevas unidades ocultas, donde para cada nueva neurona oculta, el algoritmo maximiza la magnitud de la correlación entre la nueva neurona oculta y el error residual de la red, es decir, se agregan neuronas ocultas procurando disminuir el error de la red hasta que su rendimiento sea satisfactorio.

Entonces, en una red CC no es necesario conocer *a priori* la cantidad de neuronas necesarias en la capa oculta, por tanto el aprendizaje de la red puede ser más rápido y puede tener mejor capacidad de generalización que un MLP (Villa, Velásquez, & Souza, 2008).

Figura 1-5: Esquema de aprendizaje de una red CC.



Específicamente para el caso de aplicación del pronóstico de series de tiempo, Villa *et al.* (2008) mostraron que las redes CC son una herramienta adecuada para este problema, dado que lograron resultados similares, y en algunos casos mejores, que los obtenidos por Ghiassi *et al.* (2005) con DAN2 (*Dynamic Architecture for Artificial Neural Networks*), y mejores en todos los casos que los obtenidos con redes tipo MLP, cuando se pronosticaron diversas series de tiempo ampliamente usadas en la literatura.

Desde el punto de vista de arquitectura, es posible obtener una red CC a partir de un MLP, realizando las siguientes modificaciones a la Ecuación (1.2):

- Se restringe que la función de activación de las neuronas de la capa de salida sea lineal.
- Se agregan conexiones desde las neuronas de entrada hasta la neurona de salida, esto equivale a introducir dentro del modelo una componente que es la combinación lineal de las entradas, esta modificación facilita que el modelo pueda capturar la componente lineal del conjunto de datos estudiado.
- Desde *la h-ésima* neurona de la capa oculta se agregan conexiones de salida que entran a las neuronas {*h+1,h+2,…*}, esto tiene el efecto de evitar que las

neuronas de la capa oculta puedan permutarse por lo que se reduce la multiplicidad de modelos con desempeño similar.

Consecuentemente, es posible afirmar que una red CC es una arquitectura con una mayor capacidad de aprendizaje respecto al MLP. Sin embargo, en una red CC también se presenta duplicidad de modelos por cambio de signo entre las conexiones que entran y salen de una misma neurona oculta. Tal duplicidad se controla imponiendo la restricción de que los pesos de la capa oculta a la capa de salida sólo puedan tomar valores positivo; además, ésta restricción reduce el espacio de búsqueda en la optimización de los parámetros del modelo.

Aunque, las redes CC, presentan ventajas considerables respecto a los MLP, también pueden adolecer de sobreajuste básicamente por dos razones heredadas del aprendizaje predictivo. La primera razón está relacionada con la existencia de datos extremos (*outliers*) en el conjunto de entrada, esto hace que la varianza de los parámetros de la red sea alta. La segunda con el tamaño óptimo de la red –selección adecuada de neuronas en la capa de entrada y oculta–; aunque, en las redes tipo CC no es necesario definir la cantidad de neuronas en la capa oculta, sí se requiere seleccionar cuáles son las entradas al modelo.

Si se elige una red de tamaño relativamente pequeño, ella no será capaz de generalizar con precisión los datos y, por tanto, no aprenderá sus características más importantes. Se incurre en subajuste, y en consecuencia, sería necesario aumentar el tamaño de la red. Mientras que una red de un tamaño innecesariamente grande (*i.e.,* red sobreparametrizada) tiende a aprender no sólo las características de los datos dados, sino también, el ruido y la idiosincrasia de los mismos, es decir, la red memoriza los datos de entrenamiento en vez de aprender el comportamiento de la serie; en aquel momento, la red incurre en sobreajuste y su tamaño debe ser reducido mientras se mejora su capacidad de generalización (Villa, Velásquez, & Souza, 2008).

Consecuentemente, si se presenta alguna o ambas de las causas, el modelo CC podría sobreajustar los datos, lo que puede degradar ostensiblemente su capacidad de predicción. Entonces, en la siguiente sección se exponen las razones por las cuáles la estimación de los parámetros de una RNA es un problema de optimización no-lineal mal

condicionado, con el fin de mostrar por qué es apropiado usar la metodología de regularización propuesta por Tikhonov (1973) para resolverlo parcialmente y controlar el sobreajuste.

1.3 Limitaciones del Aprendizaje

1.3.1 El aprendizaje como un problema de optimización no-lineal mal condicionado

El entrenamiento de una red neuronal puede considerarse como un problema de optimización numérica no-lineal (Haykin, 1994); para el caso de series de tiempo, tradicionalmente se minimiza la sumatoria del error cuadrático (SSE); entonces, la función objetivo, dado un conjunto de entrenamiento de tamaño *T*, está dada por:

$$F(W) = \xi_s(W) = \sum_{t=1}^{T} (\hat{y}_t - y_t)^2 \tag{1.3}$$

Básicamente, el problema se define así: minimizar la función de costo $\xi_s(W)$ respecto al vector de pesos *W* (este vector es análogo a Ω)*;* donde $\hat{y}_t$ es un modelo estimado de la Ecuación (1.2) y y_t es la serie de tiempo definida en la Ecuación (1.1). Para resolverlo existen diversas aproximaciones en la literatura; de éstas, se han usado ampliamente las basadas en gradiente (Haykin, 1999). Para resolverlo mediante técnicas de gradiente se deben tener en cuenta los siguientes aspectos (Haykin, 1999):

- *F(W)* es conocida como la función objetivo, la cual es equivalente a la función de costo denotada como $\xi_s(W)$; ésta función también es conocida como la medida estándar de rendimiento, y depende del modelo de la red y de los datos de entrada.
- La función de costo $\xi_s(W)$ debe ser continua y diferenciable respecto a un vector de pesos o parámetros *W*.
- Es un problema de optimización no-lineal sin restricciones, en el cual no se tienen ninguna restricción sobre la superficie de búsqueda; esta es una de las razones por la cual no se puede garantizar en que el *W** hallado sea un óptimo global.
- El objetivo es encontrar una solución óptima *w** que satisfaga la condición:

$$\xi_s(W^*) \leq \xi_s(W) \tag{1.4}$$

- La condición necesaria de optimalidad está dada por:

$$\nabla\xi_s(W^*) = 0 \tag{1.5}$$

- Dada una red con *m* parámetros, el operador de gradiente se define como:

$$\nabla = \left[\frac{\partial}{\partial W_1}, \frac{\partial}{\partial W_2}, \ldots, \frac{\partial}{\partial W_m}\right]^T \tag{1.6}$$

- Mientras que, $\nabla\xi_s(W^*)$ es el vector del gradiente de la función de costo:

$$\nabla\xi_s(W^*) = \left[\frac{\partial\xi}{\partial W_1}, \frac{\partial\xi}{\partial W_2}, \ldots, \frac{\partial\xi}{\partial W_m}\right]^T \tag{1.7}$$

- La mayoría de las técnicas de gradiente, se basan en la idea de realizar iteraciones locales descendentes; se comienza con un punto inicial aleatorio *W(0)*, luego con base en una regla, según la técnica, se generan una secuencia de vectores *W(1), W(2)*, …, de tal manera que la función de costo $\xi_s(W)$ disminuya en cada nueva iteración *(n + 1)*, tal que:

$$\xi_s(W(n+1)) < \xi_s(W(n)) \tag{1.8}$$

- Se espera que el algoritmo converja a una solución óptima local *W**; es posible, que la técnica converja a un óptimo local dado que la superficie del error es compleja e irregular, es decir, la superficie del error posee múltiples puntos de mínima, y puede cambiar drásticamente durante el entrenamiento (por ejemplo, al agregar una nueva neurona en la capa oculta).
- Las técnicas de optimización basadas en gradiente que comúnmente se han utilizado para entrenar redes neuronales son: *Steepest Descent*; Newton; Gauss-Newton; *Backpropagation*; *Resilient Backpropagation*; BFGS, Broyden-Fletcher-Goldfarb-Shanno; entre otros.

Sin embargo, a diferencia de la mayoría de los problemas de optimización numérica, el problema del entrenamiento de una red neuronal no se soluciona simplemente cumpliendo con la condición de optimalidad dada por la Ecuación (1.5), dado que también se deben tener en cuenta las siguientes consideraciones:

- Cuando la cantidad de parámetros de la red neuronal es cercana a la cantidad de datos de entrenamiento, el problema está sobre-determinado. Consecuentemente, la red puede ajustarse y memorizar cada uno de los detalles de los datos de entrenamiento; esto reduce notablemente la capacidad de generalización de la red (Haykin, 1994). En este orden de ideas, asumiendo que se puede encontrar un *W** tal que minimice la Ecuación (1.3) y se cumpla el criterio dado por la Ecuación (1.5), esto puede provocar que la red neuronal se ajuste perfectamente a los datos de entrenamiento, pero no garantiza que la red posea una adecuada capacidad de generalización e incurrir en sobreajuste.
- A pesar de que el problema esté sobre-determinado, puede estar correctamente planteado desde el punto de vista matemático, es decir, la función objetivo dada por la Ecuación (1.5) puede tener su correspondiente vector de gradiente dado por la Ecuación (1.7) y se puede aplicar alguna técnica basada en gradiente. Por tanto, la técnica de optimización continúa funcionando y la teoría de gradiente se cumple, a pesar de que el problema esté sobre-determinado. Es decir, la técnica de optimización es independiente del problema de sobre-determinación.
- El problema de sobre-determinación está asociado con la cantidad de parámetros de la red neuronal, es decir, con el diseño de la red neuronal (*e.g.,* cantidad de neuronas en la capa de entrada, capa oculta).

Para analizar el problema de la sobre-determinación, el entrenamiento o aprendizaje de una red neuronal puede considerarse como un problema de reconstrucción de híper-superficies, en el cual, el conjunto de datos de entrenamiento puede ser escaso para hacer un mapeo multidimensional entre las neuronas de salida y las neuronas de entrada de la red. En general, este tipo de problemas es llamado problemas inversos, los cuales pueden ser bien condicionados o mal condicionados; dado el contexto de las redes neuronales interesan los no-lineales (Haykin, 1994).

Un problema inverso no-lineal está bien condicionado sí, para una función de mapeo desconocida *F* que relaciona al dominio *X* y al rango *Y*, se cumplen las siguientes tres condiciones (Morozov V. , 1984; Morozov V. , 1993; Tikhonov & Arsenin, 1977):

1. ***Existencia***: para cada vector de entrada $x \in X$, existe una salida *y = F(x)*, donde $y \epsilon Y$.

2. ***Unicidad***: para algún par de vectores de entrada $\{x_1, x_2\} \in X$, se tiene que $F(x_1) = F(x_2)$ sí, y sólo sí, $x_1 = x_2$.
3. ***Continuidad:*** la función de mapeo es continua, esto es, si para algún $\varepsilon > 0$ existe un $\delta = \delta(\varepsilon)$ que cumpla la condición que dado $\rho_X(x, t) < \delta$ implica que $\rho_Y(F(x), F(t)) < \varepsilon$, donde $\rho(\cdot,\cdot) < \varepsilon$ es el símbolo de la distancia entre dos argumentos en sus respectivos espacios.

Entonces, el aprendizaje de una red neuronal, visto como un problema de reconstrucción de híper-superficies, es un problema inverso no-lineal mal condicionado por las siguientes razones (Haykin, 1994):

- Se pueden realizar múltiples combinaciones de neuronas de entrada y neuronas de salida, lo cual da lugar a múltiples funciones de mapeo; es decir, se pueden reconstruir diferentes superficies de mapeo utilizando los mismos datos de entrenamiento; entonces, se viola el criterio de unicidad.
- El ruido, imprecisiones y datos extremos presentes en el conjunto de entrenamiento, pueden ocasionar que se generen salidas por fuera del rango *Y* para una entrada específica del dominio *X*; con esto, se incumplen los criterios de existencia y continuidad.

1.3.2 La regularización para resolver problemas no-lineales mal condicionados y controlar el sobreajuste

Con el fin de controlar éste problema, Tikhonov en (1963) propuso la metodología de regularización para resolver problemas mal condicionados. La idea principal del método es estabilizar la solución usando algún tipo de función para penalizar la función objetivo. En general, el método de regularización tiene como objetivo realizar un intercambio equilibrado entre la fiabilidad de los datos de entrenamiento y la bondad del modelo, y por ende controlar el sobreajuste. En procedimientos de aprendizaje supervisado, el intercambio se realiza a través de la minimización del riesgo total (Haykin, 1999), dado por la expresión:

$$R(W) = \xi_s(\mathrm{W}) + \lambda \xi_c(\mathrm{W}) \tag{1.9}$$

La Ecuación (1.9) corresponde a un caso general del método de regularización de Tikhonov (1973) para solucionar problemas mal condicionados (*e.g.*, el entrenamiento de una red neuronal), en este:

- $\xi_s(W)$ se conoce como la medida estándar de rendimiento, acostumbra utilizar el error cuadrático (SSE) o el error cuadrático medio (MSE); éste término corresponde a la Ecuación (1.3), de este modo R(W) puede definirse como:

$$R(W) = \sum_{t=1}^{T} (\hat{y}_t - y_t)^2 + \lambda \xi_c(\mathrm{W}) \tag{1.10}$$

- $\xi_c(w)$ es la penalización compleja, también conocida como estrategia de regularización o término de regularización, que para una red en general, puede definirse como la integral de suavizado de orden *k* (Haykin, 1999):

$$\xi_c(w,k) = \frac{1}{2} \int \left\| \frac{\partial^k}{\partial w^2} F(w,m) \right\|^2 \mu(w) dw \tag{1.11}$$

- En la Ecuación (1.11), $F(w,m)$ es el mapeo de entrada–salida realizado por el modelo, $\mu(w)$ es alguna función de ponderación que determina la región del espacio de entrada sobre la cual la función $F(w,m)$ es requerida para ser suavizada.
- λ es el parámetro o factor de regularización que controla el nivel de incidencia de $\xi_c(w)$ sobre el entrenamiento de la red, en secciones posteriores de éste documento se discutirá sobre este factor.

Dada la Ecuación 1.10, desde el punto de vista de optimización no-lineal numérica, el método de regularización es una especie de penalización que se impone sobre la función objetivo definida en la Ecuación 1.3. A pesar de que la idea principal del método es estabilizar la solución usando algún tipo de función para penalizar la función objetivo (*i.e.*, usar una función o estrategia de regularización), su aplicación es compleja, dado que, el problema no es solamente seleccionar una determinada estrategia de regularización entre las disponibles, sino que también es necesario determinar qué tanto debe incidir tal

estrategia sobre el entrenamiento de la red. A continuación se describen dos funciones de penalización compleja utilizadas tanto en los MLP como en las redes CC.

- **La descomposición de pesos (DP) – (Weight Decay)**

El procedimiento de descomposición de pesos propuesto por Hinton (1989), opera sobre algunos pesos sinápticos de la red forzándolos a tomar valores cercanos a cero y permitiendo a otros conservar valores relativamente altos. Esta discriminación permite agrupar los pesos de la red en: pesos que tienen poca o ninguna influencia sobre el modelo; y pesos que tienen influencia sobre el modelo, llamados pesos de exceso. Para ésta estrategia el procedimiento la penalización de complejidad se define como:

$$\xi_c(w) = \left\|w_{p,h}\right\|^2 = \sum_{h=1}^{H}\sum_{p=1}^{P} w_{p,h}^2 \tag{1.12}$$

En la Ecuación 1.12, $w_{p,h}$ son los pesos de la entrada *p* a la neurona *h*, es decir, los pesos entre la capa de entrada y la oculta. El tratamiento de los pesos de la red CC es similar al de los MLP; todos los pesos son tratados igual, es decir, se parte del supuesto que la distribución de los pesos en el espacio estará centrada en el origen.

La descomposición de pesos es una de las estrategias de regularización más utilizadas en la literatura (Leung, Wang, & Sum, 2010); dado que su implementación es computacionalmente sencilla, no depende de parámetros adicionales y permite mejorar la capacidad de generalización de la red neuronal.

Por otro lado, en problemas de ajustes de curvas también se le conoce con el nombre de regresión de borde (*ridge regression*) (Bishop, 1994), porque su efecto es similar a la técnica de regresión del mismo nombre propuesta por Hoerl y Kennard (1970). Además, en aprendizaje Bayesiano, es posible hallar la correspondiente función de distribución de probabilidad para esta estrategia, la cual depende tanto de los pesos como de su agrupación (neuronas, también llamadas hiperparámetros) (Bishop, 1994).

Sin embargo, una de las limitaciones de esta estrategia es que puede ser inconsistente con ciertas propiedades de escalado de los pesos en el mapeo entre entrada y salida de la red, para más detalles véase a Bishop (1994). Pero tal vez la mayor dificultad, es que

una vez se aplica esta estrategia muchos parámetros de la red tienden a ser pequeños respecto a otros, pero no se puede tener certeza de cuáles eliminar, y puede que no contribuya significativamente a la solución del problema de sobreajuste (Bishop, 1994). Para solventar en cierta medida este inconveniente, Weigend *et al.* (1991) propuso la estrategia de eliminación de pesos, la cual se describe a continuación.

- **La eliminación de pesos (EP) – (Weight Elimination)**

Este método de regularización descrito por Weigend *et al.* (1991) define la penalización de complejidad como:

$$\xi_c(w) = \sum_{h=1}^{H} \sum_{p=1}^{P} \frac{\left(w_{p,h}/w_0\right)^2}{1 + \left(w_{p,h}/w_0\right)^2} \tag{1.13}$$

En la Ecuación 1.13, w_0 es un parámetro predefinido, el cual se elige según el criterio del experto. El término $w_{p,h}/w_0$ hace que la penalización tenga un comportamiento simétrico. Además, cuando $|w_{p,h}| \lll w_0$, $\xi_c(w)$ tiende a cero, es decir, para el aprendizaje el peso sináptico $w_{p,h}$ es poco fiable, por consiguiente puede ser eliminado de la red. Mientras que cuando$|w_{p,h}| \ggg w_0$, $\xi_c(w)$ tiende a uno, entonces el peso $w_{p,h}$ es importante para el proceso de aprendizaje. En conclusión, éste método busca los pesos que tienen una influencia significativa sobre la red, y descarta los demás.

Es evidente que éste método es más complejo de aplicar, dado que es necesario elegir un w_0 tal que permita eliminar algunos pesos, lo cual se suma a la dificultad de la selección del factor λ de regularización.

1.4 Avances en el control del sobreajuste en redes neuronales artificiales

Uno de los aportes de ésta sección es aplicar la metodología de la revisión sistemática de la literatura (SLR – systematic literature review) presentada por Kitchenham (2004) con el fin de identificar:

- Otro tipo de soluciones para el control del sobreajuste en redes neuronales artificiales, particularmente, redes CC.

- Cuáles son los aportes más relevantes en cuanto al control del sobreajuste en redes neuronales tipo CC.
- Problemas pendientes por resolver para controlar efectivamente el sobreajuste en las redes neuronales tipo CC.
- Los problemas más relevantes de la aplicación de la regularización para controlar el sobreajuste.

1.4.1 Preguntas de investigación

Las preguntas de investigación (PI) de esta revisión sistemática de la literatura son:

- PI1: ¿Cuáles mecanismos se han propuesto en la literatura para controlar el sobreajuste en redes neuronales artificiales?
- PI2: ¿Cuáles son las estrategias de regularización que se han propuesto en la literatura para controlar el sobreajuste?
- PI3: ¿Cuáles son las estrategias de regularización que se han propuesto y usado en la literatura para controlar el sobreajuste en redes tipo CC?
- PI4: ¿Cuáles son los problemas más relevantes para utilizar algún mecanismo específico para el control del sobreajuste?

1.4.2 Proceso de Búsqueda

La búsqueda se ha realizado manualmente en el sistema de indexación SCOPUS, los criterios de búsqueda (CB) que se usaron son:

- Primer criterio de búsqueda (CB1):
  - Consulta: (TITLE-ABS-KEY("Neural Networks") AND TITLE-ABS-KEY((overfitting OR ill-posed))) AND DOCTYPE(ar OR re)
  - Resultado total de documentos: 513
  - Documentos seleccionados, de acuerdo a los criterios definidos en la Sección 1.4.3
- Segundo criterio de búsqueda (CB2):
  - Consulta: TITLE-ABS-KEY("neural network" AND regularization) AND DOCTYPE(ar OR re)
  - Resultado total de documentos: 598

  - Documentos seleccionados, de acuerdo a los criterios definidos en la Sección 1.4.3: 3

- Tercer criterio de búsqueda (CB3):
  - Consulta: (TITLE-ABS-KEY("Cascade Correlation") AND TITLE-ABS-KEY(overfitting OR regularization)) AND DOCTYPE(ar OR re)
  - Resultado total de documentos: 9
  - Documentos seleccionados, de acuerdo a los criterios definidos en la Sección 1.4.3: 2

1.4.3 Criterios de inclusión o exclusión

El criterio determinante de inclusión fue seleccionar los artículos o estudios que describen el problema del sobreajuste para modelos de redes neuronales artificiales o detallan algún mecanismo para controlarlo. Estos artículos fueron revisados y permiten responder las preguntas de investigación planteadas en la Sección 1.4.1.

Los artículos con las siguientes características fueron excluidos:

- Los que no describen el problema del sobreajuste para modelos de redes neuronales artificiales.
- Los que presentan una aplicación de las estrategias de regularización sin un desarrollo teórico o que no definen un objetivo o pregunta de investigación.
- Los que replican o simplemente aplican estrategias de regularización en casos de aplicación.
- Los que han sido poco citados (menos de 3 citaciones). Esto aplica para los que han sido publicados antes del 2008.

1.4.4 Criterios de evaluación (CE)

Cada artículo fue evaluado de acuerdo a los siguientes criterios, y cada uno se calificará de acuerdo a una escala cualitativa:

- CE1: ¿Describe el problema del sobreajuste para modelos de redes neuronales?
  - Sí (S), la descripción del problema se realiza de manera explícita en el artículo.
  - Parcialmente (P), la descripción del problema se realiza de manera implícita citando a otros autores.
  - No (N), carece de la descripción del problema.

- CE2: ¿Describe algún mecanismo para controlar el sobreajuste de redes neuronales?
  - Sí (S), describe explícitamente algún mecanismo para controlar el sobreajuste.
  - Parcialmente (P), describe implícitamente algún mecanismo citando a otros autores.
  - No (N), carece de la descripción de algún mecanismo.

- CE3: ¿Describe las dificultades de usar algún mecanismo específico para controlar el sobreajuste en redes neuronales?
  - Sí (S), describe explícitamente las dificultades de aplicar algún mecanismo para controlar el sobreajuste.
  - Parcialmente (P), describe implícitamente las dificultades de aplicar algún mecanismo citando a otros autores.
  - No (N), carece de la descripción la descripción de problemas para aplicar algún mecanismo para el control del sobreajuste.

- CE4: ¿Dado un conjunto de mecanismos para controlar el sobreajuste, se recomienda cuál es el más adecuado para algún tipo de red neuronal específico?
  - Sí (S), recomienda y justifica explícitamente cuál mecanismo de control del sobreajuste se debe usar con algún tipo específico de red neuronal.
  - Parcialmente (P), no recomienda de manera clara y precisa cuál mecanismo de control del sobreajuste se debe usar con algún tipo específico de red neuronal
  - No (N), no recomienda, ni justifica cuál mecanismo de control del sobreajuste se debe usar con algún tipo específico de red neuronal

- CE5: ¿Describe algún mecanismo para controlar el sobreajuste de redes neuronales tipo CC?
  - Sí (S), describe explícitamente algún mecanismo para controlar el sobreajuste en redes tipo CC.
  - Parcialmente (P), describe implícitamente algún mecanismo citando a otros autores para controlar el sobreajuste en redes tipo CC.
  - No (N), carece de la descripción de algún mecanismo para controlar el sobreajuste en redes tipo CC.

1.4.5 Análisis cuantitativo de la revisión de la literatura

Los resultados obtenidos mediante los criterios de búsqueda (CB) definidos en la Sección 1.4.2 se resumen en la Tabla 1-2. también, muestra el resultado de los criterios de evaluación (CE) definidos en la Sección 1.4.4. Adicionalmente, se presenta una columna con el mecanismo del control de sobreajuste que se menciona en el respectivo artículo. De acuerdo a los criterios de búsqueda y de evaluación, se encontraron 10 artículos relevantes, de los cuales:

- El 60% cumple totalmente con el CE1, es decir, explica explícitamente el problema del sobreajuste. El restante no lo cumple.
- El 90% cumple con el CE2, es decir, describe algún mecanismo para controlar la regularización. El restante no lo cumple.
- El 60% cumple con el CE3, es decir, de alguna manera describen algunas dificultades de aplicar o usar algún mecanismo de regularización. El restante no lo cumple.
- El 30% cumple parcialmente con el CE4, es decir, no recomiendan de manera clara y sencilla algún mecanismo de regularización. El restante no lo cumple.
- El 20% cumple con CE5, es decir, describe algún mecanismo para controlar el sobreajuste en las redes CC.

Tabla 1-2: Evaluación de la calidad de los documentos encontrados.

	Artículo	CB	CE1	CE2	CE3	CE4	CE5	Mecanismo de Control del Sobreajuste
1.	(Giustolisi & Laucelli, 2005)	1	S	P	P	N	N	▪ Penalización: regularización (Tikhonov A. , 1973; Haykin, 1999). ▪ División del conjunto de entrenamiento: parada temprana (Lang et al.,1990); validación cruzada (Haykin, 1999). ▪ Control de la cantidad de pesos: suavizado de pesos compartidos (Nowlan & Hinton, 1992); daño óptimo cerebral (Le Cun, Denker, & Solla, 1990)
2.	(Hawkins, 2004)	1	S	N	N	N	N	
3.	(Leung, Wang, & Sum, 2010)	1	P	P	S	P	N	Regularización: Descomposición de pesos. (Hinton, 1989)
4.	(Schittenkopf, Deco, & Brauer, 1997)	1	S	P	N	N	N	▪ Control de la cantidad de pesos (Schittenkopf, Deco, & Brauer, 1997). ▪ Penalización: regularización (Tikhonov A. , 1973; Haykin, 1999)
5.	(Karystinos & Pados, 2000)	1	S	S	S	P	N	▪ Regularización: (Tikhonov A. , 1973; Haykin, 1999) ▪ Validación cruzada (Haykin, 1999). ▪ Suavizado de pesos compartidos (Nowlan & Hinton, 1992)
6.	(Burger & Neubauer, 2003)	2	P	S	P	N	N	Regularización: descomposición de pesos. (Hinton, 1989); suavizado aproximado (Moody & Rögnvaldsson, 1997)
7.	(Leung & Chow, 1999)	2	S	P	S	N	N	Regularización: descomposición de pesos (Hinton, 1989).
8.	(Setiono, 1997)	2	S	S	P	P	N	Regularización: eliminación de pesos (Weigend *et al.*, 1991).
9.	(Tetko & Villa, 1997)	3	P	S	N	N	S	Regularización: descomposición de pesos (Hinton, 1989).
10.	(Xu & Nakayama, 1997)	3	P	P	P	N	S	Regularización: descomposición de pesos (Hinton, 1989); eliminación de pesos (Weigend *et al.*, 1991); olvido de pesos (Ishikawa, 1989; Ishikawa, 1996); suavizado aproximado (Moody & Rögnvaldsson, 1997).

1.4.6 Respuesta preguntas de investigación

De acuerdo a la SLR y con base en los artículos seleccionados, ver Tabla 1-2, a continuación se responden las preguntas de investigación:

1. PI1: ¿Cuáles mecanismos se han propuesto en la literatura para controlar el sobreajuste en redes neuronales artificiales?
    - Según Giustolisi y Laucelli (2005), los mecanismos de regularización se pueden agrupar en las siguientes tres categorías:
        1. Penalización de la función de costo, regularización (Tikhonov A. , 1973; Haykin, 1999)
            - Descomposición de Pesos (Hinton, 1989).
            - Eliminación de Pesos (Weigend, Rumelhart, & Huberman, 1991).
            - Olvido de Pesos (Ishikawa, 1989; Ishikawa, 1996).
            - Suavizado Aproximado (Moody & Rögnvaldsson, 1997).
        2. División del conjunto de entrenamiento:
            - Parada temprana (Lang, Waibel, & Hinton, 1990)
            - Validación cruzada (Haykin, 1999)
        3. Control de la cantidad de pesos:
            - Suavizado de pesos compartidos (Nowlan & Hinton, 1992)
            - Daño óptimo cerebral (Le Cun, Denker, & Solla, 1990)

2. PI2: ¿Cuáles son las estrategias de regularización que se han propuesto en la literatura para controlar el sobreajuste?
    - De acuerdo a los trabajos citados en la Tabla 1, a excepción del 2, se han propuesto las siguientes cuatro estrategias de regularización, las cuales se han utilizado en estudios relevantes:
        1. Descomposición de Pesos (Hinton, 1989).
        2. Eliminación de Pesos (Weigend, Rumelhart, & Huberman, 1991).
        3. Olvido de Pesos (Ishikawa, 1989; Ishikawa, 1996).
        4. Suavizado Aproximado (Moody & Rögnvaldsson, 1997).

3. PI3: ¿Cuáles son las estrategias de regularización que se han propuesto y usado en la literatura para controlar el sobreajuste en redes tipo CC?
    - Según Xu y Nakayama (1997) se han usado las siguientes estrategias de regularización:
        - Descomposición de Pesos (Hinton, 1989).
        - Eliminación de Pesos (Weigend, Rumelhart, & Huberman, 1991).
        - Olvido de Pesos (Ishikawa, 1989; Ishikawa, 1996).
        - Suavizado Aproximado (Moody & Rögnvaldsson, 1997).

4. PI4: ¿Cuáles son los problemas más relevantes para utilizar algún mecanismo específico para el control del sobreajuste?
    - Según Leung *et al.* (2010), Xu y Nakayama (1997) y Setiono (1997) una de las principales dificultades de aplicar las estrategias de regularización es la selección del parámetro λ.
    - Según Karystinos y Pados (2000), una de las dificultades de usar mecanismos que dividan el conjunto de entrenamiento (por ejemplo, validación cruzada) es que si se posee un conjunto de entrenamiento relativamente pequeño, se puede perder información importante para la estimación de los parámetros; es decir, la decisión de cómo dividir el conjunto de entrenamiento (en un subconjunto de entrenamiento y otro de prueba) es crítica, dado que es posible que no se seleccione la cantidad adecuada de datos para la prueba y se desperdicien para el entrenamiento.

Entonces, con base en la revisión sistemática de la literatura, en la siguiente sección se plantea una discusión alrededor de la complejidad de aplicar la regularización para controlar el sobreajuste en redes CC.

1.5 Discusión sobre la complejidad de la aplicación de la regularización

La regularización en términos de la Ecuación (1.10) puede solucionar el problema del sobreajuste en las redes neuronales; sin embargo, para que su aplicación sea efectiva es necesario tomar dos decisiones que son críticas y subjetivas: la primera es seleccionar la estrategia de regularización $\xi_c(w)$; y la segunda es determinar el valor del parámetro de

regularización λ (Evgeniou, Poggio, Pontil, & Verri, 2002; Hagiwara, 2002; Chow & Cho, 2007).

La selección de la estrategia de regularización depende del efecto que se desee obtener sobre los parámetros o pesos de la red; sin embargo, tanto la descomposición de pesos como la eliminación de pesos tienen problemas conceptuales y prácticos que fueron descritos en la Sección 1.3.2 y no garantizan que se controle efectivamente el sobreajuste, dado que no tienen en cuenta la varianza o desviación estándar de los datos ni la de los parámetros. Además, en la literatura se ha mostrado experimentalmente que tales estrategias ayudan a controlar el problema en las redes CC siempre y cuando se seleccione un λ adecuado (Xu & Nakayama, 1997; Villa & Velásquez, Regularización de Redes Cascada Correlación con Regresión en Cadena, 2010); Chow y Cho (2007) muestran que la efectividad de ambas estrategias es significativamente afectada por la selección de λ al pronosticar con un MLP una serie de tiempo.

Asumiendo que la estrategia de regularización es la adecuada y tiene el efecto esperado, el parámetro λ representa el nivel de incidencia que ésta tendrá en el entrenamiento de la red. Si λ es cero, el aprendizaje se realiza sin la penalización; evidentemente, la estrategia seleccionada no tendrá incidencia sobre el aprendizaje y éste se realizará sin restricción alguna; mientras que, si λ tiende a infinito el aprendizaje sólo se basará en la penalización impuesta, suavizando más de lo necesario la curva del error, tendiendo los parámetros a cero y evitando que la red generalice adecuadamente los datos de entrenamiento (Bishop, 1994; Haykin, 1999; Chow & Cho, 2007). Cuando se selecciona un valor adecuado de λ, la penalización permitirá generalizar adecuadamente los datos de entrenamiento (Chow & Cho, 2007).

Es evidente que la selección de la técnica de regularización y la del parámetro, son dos decisiones que están íntimamente ligadas, dado que cuando se selecciona adecuadamente un valor de λ, las estrategias de regularización permiten controlar de manera efectiva el problema del sobreajuste, y permiten entrenar la red para que generalice los datos de entrenamiento. Si no se selecciona un valor adecuado para λ, las estrategias son relativamente sensibles al ruido presente en los datos de entrenamiento (Chow & Cho, 2007).

Desde el punto de vista de la teoría de aprendizaje estadístico, la correcta elección del valor de λ permite convertir un problema mal condicionado en uno correctamente condicionado (Evgeniou, Poggio, Pontil, & Verri, 2002). Matemáticamente, λ y la estrategia de regularización son en conjunto la penalización sobre la curva de solución (error de entrenamiento), si λ es muy alta esta curva se suavizará demasiado; si por el contrario λ tiende a cero la curva puede que no se suavice lo suficiente; en ambos casos se busca un λ adecuado para suavizar la curva de tal manera que ayude al algoritmo de optimización a encontrar una mejor solución que la hallada cuando λ es igual a cero. En este caso λ también es conocido como el grado de regularización que controla el grado de suavizamiento de la solución. Entonces, la regularización permite estabilizar la solución de un problema mal condicionado, suavizando la función de costo o función objetivo, siempre y cuando se seleccione un λ adecuado (Morozov V. , 1984).

Este problema ha sido discutido por Evgeniou *et al.* (2002) para las máquinas de vectores de soporte y para las redes con función de base radial, específicamente para el caso de la clasificación de patrones, teniendo en cuenta que los nodos, unidades o parámetros de estos modelos no aumentan durante el aprendizaje, lo que sí ocurre en las redes CC; Vapnik (1998) lo hizo desde el punto de vista del aprendizaje estadístico; mientras que Hagiwara (2002) encontró un λ teórico para redes lineales en el contexto de la regresión estadística. Desde un punto de vista práctico son pocas las evidencias encontradas, Villa *et al.* (2008) experimentaron con redes CC para pronosticar series de tiempo, y recomendaron algunos valores discretos para λ.

En general, la elección del valor de λ es a menudo dependiente de los datos de entrenamiento, y es necesario determinarlo experimentalmente (Chow & Cho, 2007). Desafortunadamente, en la práctica la elección de este parámetro no es sencilla dado que:

- Su dominio es (0, +∞) (Haykin, 1999), lo cual es un espacio de búsqueda infinito, en el cual entre más se tiende a cero menos incidencia tendrá la regularización, por el otro lado, entre más se tiende a infinito más incidencia tendrá. Adicionalmente, se ha mostrado de forma experimental que λ puede ser un parámetro muy sensible. (Villa, Velásquez, & Souza, 2008; Chow & Cho, 2007; Leung & Chow, 1999).

- Es complejo encontrar simultáneamente un valor para λ mientras se agregan dinámicamente neuronas en la capa oculta de la red CC (Xu & Nakayama, 1997). También, agrega complejidad al problema, el hecho de seleccionar los rezagos (neuronas en la capa de entrada) mientras se busca simultáneamente un valor para λ. Y es aún más complejo, buscar y encontrar dinámicamente: cuáles son los rezagos; cuánta es la cantidad de neuronas en la capa de entrada; y cuál es el valor de λ para una determinada estrategia de regularización.

- Minimizar la combinación del error empírico de los datos de entrenamiento junto con el factor de penalización es una tarea compleja, dado que se debe controlar la capacidad de suavización de las funciones de penalización (estrategias de regularización) (Evgeniou, Pontil, & Poggio, 1999).

- Este parámetro controla el balance entre el sesgo y la varianza del error de generalización esperado (Hagiwara, 2002). De ahí, que su selección sea uno de los mayores problemas en el uso de las estrategias de regularización y sea determinante en el rendimiento de la red neuronal, especialmente en su capacidad de generalización (Chow & Cho, 2007).

- Se requiere un λ diferente dependiendo de la estrategia de regularización seleccionada; el problema se puede complicar más si la estrategia requiere parámetros adicionales, como la estrategia de Eliminación de Pesos dada por la Ecuación (1.13), en la cual también es necesario determinar w_0.

- La elección inapropiada de λ puede deteriorar la capacidad de generalización de la red (Palit & Popovic, 2005); por tanto, Weigend *et al.* (1991) recomiendan actualizar su valor durante el entrenamiento de la red, pero no especifican cómo se debe hacer.

- Para el caso particular del pronóstico de series de tiempo, a diferencia de la regresión, se debe tener en cuenta el orden de los datos, así como las propiedades estadísticas que este ordenamiento induce sobre la información. Además, las experiencias reportadas en la literatura muestran que la dinámica de

la mayoría de las series de tiempo es no lineal, cambiante en el tiempo y puede poseer una gran cantidad de variables explicativas (Velásquez, Dyner, & Souza, 2005). De ahí, un λ hallado para la regresión puede no ser válido para series de tiempo.

- No existen criterios en la literatura más relevante, sobre cómo elegir objetivamente tanto la estrategia de regularización como la magnitud de λ para las redes cascada-correlación.

En conclusión, la búsqueda del valor adecuado para λ no es una tarea trivial y es crucial para el entrenamiento y control del sobreajuste de una red neuronal, especialmente para el caso de las redes CC. Con base en la revisión sistemática de la literatura y en discusión realizada, surgen diversas preguntas que se plantean en la siguiente sección.

1.6 Preguntas emergentes

Con base en la revisión de la literatura e implicaciones expuestas en las secciones anteriores es posible extraer los siguientes problemas puntuales:

- No existen razones metodológicas o teóricas para seleccionar una estrategia de regularización específica entre varias alternativas para las redes Cascada Correlación.
- No existe una vía sistemática que permita seleccionar cuáles son los parámetros de exceso que tienen poca influencia sobre un modelo de red Cascada Correlación o de controlar efectivamente su magnitud.
- No hay claridad acerca de las implicaciones metodológicas, conceptuales y prácticas que acarrea la aplicación de estrategias de regularización en las redes neuronales artificiales.
- No existe una vía sistemática aceptada en la literatura para determinar el valor λ con una estrategia de regularización determinada, para el caso de las redes neuronales tipo Cascada Correlación.
- No existe una vía sistemática que permita determinar bajo qué condiciones el comportamiento esperado de una estrategia de regularización se cumple.

- No hay claridad acerca de las relaciones existentes entre la aplicación de una estrategia de regularización para una red neuronal y las características propias de la serie de tiempo.
- No existe una vía sistemática, a través de la regularización, que permita establecer dinámicamente cuáles son las neuronas de entrada (rezagos) que son innecesarias en una red Cascada Correlación.

Los problemas aquí planteados se manifiestan como necesidades que requieren ser cubiertas con el fin de obtener una estrategia sistemática para la especificación y construcción de un modelo de red Cascada-Correlación regularizada para el pronóstico de series de tiempo. En esta investigación se apunta la resolución de dichos problemas, especialmente orientadas a la consecución de aportes metodológicos y conceptuales. Consecuentemente, en la siguiente sección se plantean los objetivos de ésta investigación.

1.7 Objetivos de la Tesis

En la sección anterior se han enunciado aspectos fundamentales que deberían tenerse en cuenta para que se den avances metodológicos, teóricos, conceptuales y prácticos en la predicción de series de tiempo con redes CC. A partir de ellos, se establecen objetivos de investigación de nivel doctoral a continuación.

1.7.1 Objetivo General

Diseñar un criterio para la selección del modelo de red neuronal cascada correlación que permita controlar integralmente el tamaño del conjunto de entrenamiento, la complejidad del modelo y la magnitud de pesos para obtener modelos con buena capacidad de generalización.

1.7.2 Objetivos Específicos

1. Determinar un criterio de información para las redes cascada correlación para la selección de modelos entre un conjunto finito.

2. Determinar una estrategia sistemática de selección del parámetro de regularización para la red neuronal cascada correlación que permita controlar la complejidad del modelo.
3. Determinar una estrategia de regularización para la red neuronal cascada correlación que permita controlar la magnitud de los parámetros del modelo.
4. Diseñar un criterio integral con base en los elementos definidos en los Objetivos Específicos 1, 2 y 3.

1.8 Aportes y contribuciones

Ésta investigación es una contribución tanto conceptual como metodológica a los problemas de la predicción de series de tiempo. En esta se proponen nuevas aproximaciones a dicho problema, tales como: la utilización de las redes tipo cascada correlación para modelar y pronosticar series de tiempo; la incorporación de técnicas de regularización en la arquitectura de las redes CC con el fin de controlar el problema del sobreajuste; la utilización de redes CC regularizadas para realizar el modelado y pronóstico de series de tiempo; y la descripción de un protocolo de selección de una red CC para el pronóstico de series de tiempo.

Además, tiene un impacto en el área asociado al desarrollo de nuevos conocimientos tanto en el área de la inteligencia computacional como en el área de la predicción. Su impacto en el área de inteligencia computacional está relacionado con el desarrollo de una estrategia de especificación para el tipo de red estudiado. El área de la predicción se ve impactada por la incorporación de la red Cascada-Correlación al conjunto de técnicas usada para el modelado y la predicción de series de tiempo, así como por la aplicabilidad práctica que este tiene. Ello es debido a que hay un valor agregado relacionado con la utilidad del modelo para representar la dinámica de series financieras y económicas, y las aplicaciones posteriores de estos, como por ejemplo, en el modelado y pronóstico de precios de electricidad.

También tiene un impacto importante al interior del grupo de investigación, ya que fortalece el área de predicción, genera una experiencia muy importante en el modelado de series de tiempo, y de inteligencia computacional. Igualmente, fortalece el grupo de investigación al darle continuidad a su trabajo, y da pie a la formulación de nuevos proyectos en el mediano plazo.

Desde la práctica, todos aquellos agentes del mercado interesados en el modelado de series de tiempo se ven impactados, ya que se fortalece el uso de esta clase de modelos, a partir de los cuales se pueden mejorar sus procesos de decisión al contar con información de mayor calidad.

1.9 Mapa del documento

Con el fin de cumplir los objetivos propuestos en esta Tesis, en el Capítulo 1 se han descrito los aspectos teórico-conceptuales del aprendizaje predictivo, tanto desde el punto de vista de marco de trabajo como desde el de la arquitecturas de perceptrón multicapa y redes cascada-correlación; además, se describen las limitaciones de este tipo de aprendizaje, donde la más sobresaliente es el sobreajuste, y se describen las propuestas de solución que se han dado tradicionalmente en la literatura, finalmente se esboza que tales soluciones no controlan eficazmente el problema. De las soluciones propuestas, la más formal es la de Regularización, en el Capítulo 2 se discute desde los puntos de vista teórico, conceptual y experimental cómo se debe controlar el sobreajuste, por qué es difícil controlarlo y por qué la regularización no lo controla eficazmente.

Uno de los principales problemas es cómo seleccionar el parámetro de regularización, para lo cual se propone una solución descrita y evaluada en el Capítulo 3, la cual también permite usar la regularización como criterio de información. Sin embargo, hasta este punto no se ha solucionado el problema del controlar efectivamente la magnitud de los parámetros (*i.e.* Estabilidad del Modelo), con el fin de tener un criterio de información integral, en el Capítulo 4, se propone, muestra y evalúa usar la varianza y la desviación estándar como estrategias de regularización; ahora, con el criterio completo se pronostican siete series de tiempo obteniendo resultados superiores en generalización a otros modelos tradicionales como DAN2, MLP y ARIMA. Finalmente, en las conclusiones se evidencia el cumplimiento de los objetivos de esta Tesis.

2. Capítulo 2: El problema del sobreajuste, una visión complementaria

En este Capítulo se presenta la evidencia experimental del fenómeno del sobreajuste y su control con un ejercicio tradicional de ajuste de curvas con polinomios, el cual es análogo al problema que se presenta en las redes neuronales; además, se describen, evalúan y analizan experimentalmente los síntomas del sobreajuste en redes CC y su control con estrategias de regularización de decaimiento y eliminación de pesos.

2.1 El fenómeno del sobreajuste y su control con regularización

La finalidad de entrenar una red neuronal para pronosticar una serie de tiempo es representar mediante un modelo matemático el proceso generador de los datos; nunca se debe buscar una reproducción exacta de los mismos. Entonces, la meta es buscar una red que exhiba una buena capacidad de generalización, es decir, una red que tenga la capacidad de predecir el comportamiento de nuevos datos.

Una analogía simple a este proceso puede ser vista desde el uso de polinomios para el ajuste de curvas, donde un grado bajo del polinomio (pocos coeficientes) conduce a una pobre interpolación ante nuevos datos (subajuste), esto es una pobre generalización, ya que el polinomio tiene poca flexibilidad. Por otro lado, un polinomio de grado alto (muchos coeficientes) también conduce a una pobre generalización ya que se ajusta mucho al ruido en los datos (sobreajuste) (Bishop, 1995; Chow & Cho, 2007). El número de coeficientes en el polinomio controla la flexibilidad efectiva o complejidad del modelo. Análogamente, las redes neuronales de gran tamaño son propensas a aprender las particularidades o ruido presente en los datos de entrenamiento y a incurrir en el problema conocido del sobreajuste (Haykin, 1999).

A continuación, con el fin de responder a la pregunta ¿En la práctica cuáles son los síntomas del fenómeno del sobreajuste?, en ésta sección se mostrará experimentalmente los efectos de tal fenómeno, para ello se utilizará como modelo un polinomio de grado *m* y parámetros *w* dado por la ecuación:

$$y(x) = \sum_{j=0}^{m} w_j x^j \tag{2.1}$$

para ajustar N puntos de la función planteada por Bishop (1995):

$$h(x) = 0.5 + 0.4\, sin(2\pi x) + \ ruido \tag{2.2}$$

En la Ecuación (2.2), el ruido se toma como gaussiano con varianza 0,05. Análogo a los experimentos realizados por Bishop (1995), para el ajuste se utilizan 10 puntos, mientras que para interpolación 100; el ajuste se realiza minimizando la raíz cuadrada del error cuadrático medio (RMSE).

En la Figura 2-1, se evidencia el fenómeno del subajuste, esto es el otro extremo al sobreajuste, se realiza el ajuste de un polinomio de primer grado; se observa que este no es suficiente para ajustarse a los datos, dado que no es flexible, es decir, una línea recta no es adecuada para representar a los datos; mientras, que un polinomio de segundo grado tampoco resulta adecuado para representar la función *h(x)*.

Uno de tercer grado, ver Figura 2-2, visualmente se aproxima al comportamiento de la función, e intuitivamente se puede afirmar que es el modelo adecuado para representar la dinámica de los datos. Uno de cuarto grado, tiene un comportamiento similar al de tercer grado y también podría ser un modelo adecuado. Sin embargo, se puede observar que el entrenamiento se ve altamente influenciado por el primer y último dato, como también por el dato extremo. El caso de las redes neuronales, es evidentemente más complejo, ya que no se conoce la función o el proceso generador de la serie de tiempo, la elección de la cantidad de neuronas en la capa de entrada y salida no es sencilla, y la serie puede contener uno o varias observaciones atípicas o extremas.

Figura 2-1: Ajuste de h(x) con Polinomio grado 1.

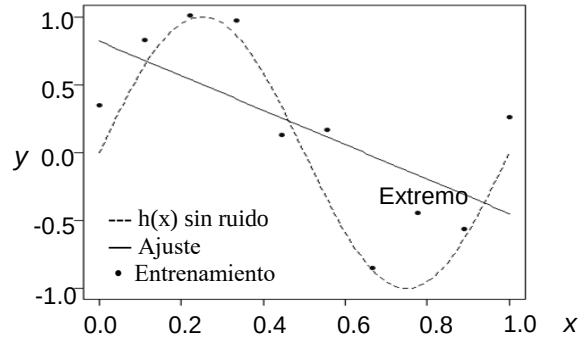
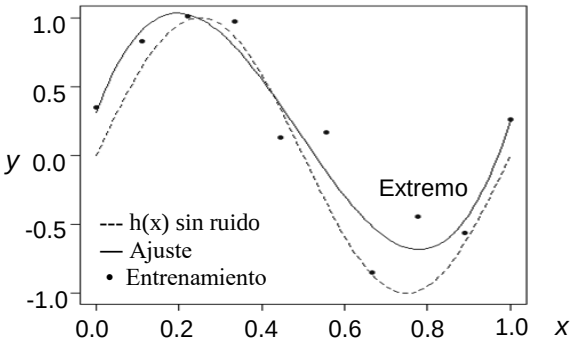
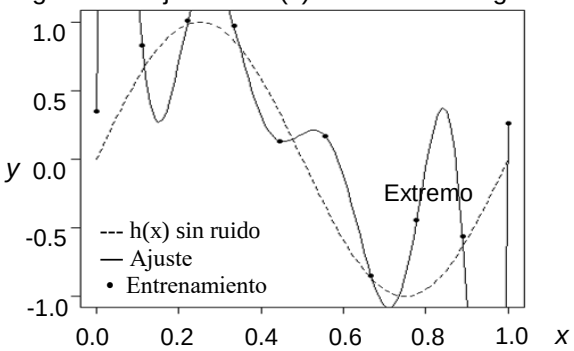


Figura 2-2: Ajuste de h(x) con Polinomio grado 3.



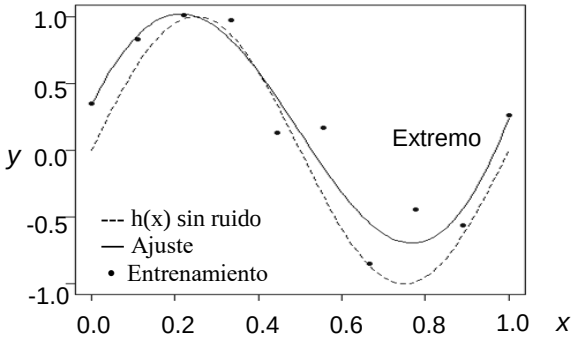
Si se continúa aumentando el grado del polinomio, se incrementa la cantidad de parámetros del modelo, cuando se llega a nueve, se tienen igual grados de libertad y el mismo se ajusta perfectamente a los datos (error de entrenamiento igual a cero), pero no se aproxima al comportamiento de la función h(x) (sobreajusta los datos), como se observa en la Figura 2-3.

Figura 2-3: Ajuste de h(x) con Polinomio grado 9.



Por otro lado, según la Figura 2-4, el modelo con mejor capacidad de generalización es el polinomio de grado cinco, donde el error de interpolación es mínimo; sin embargo, al observar la Figura 2-4, el comportamiento del polinomio de grado cinco es similar al polinomio de grado tres (Figura 2-2); en este caso es complejo decidir *a priori* cuál modelo es mejor, a pesar de que el comportamiento del modelo de grado cinco es similar al de grado tres, el primero tiene más parámetros por tanto es más complejo; si se tiene en cuenta el principio de la parsimonia el modelo adecuado es el más simple, es decir el de grado tres; sin embargo, en términos del RMSE de interpolación el modelo adecuado es el de grado cinco, porque es 60,85% menor respecto al modelo de grado tres. Este es un caso particular del dilema conocido como la maldición de la dimensionalidad (Bishop, 1995). En términos de una red neuronal, el problema es aún más complejo, pues se deben tener en cuenta los parámetros relativos a cada neurona de entrada y oculta que se incluya en el modelo; por lo cual se tiene un gran número de combinaciones que dificultan la selección adecuada del modelo (Sánchez & Velásquez, 2010).

Figura 2-4: Ajuste de h(x) con Polinomio grado 5.



La capacidad de generalización de los polinomios se mide mediante el RMSE, al interpolar 100 datos con cada uno de los polinomios estimados. En la Figura 2-5, se presenta tanto el error de entrenamiento como el de interpolación, incrementando el grado del polinomio. Se evidencia que a medida que se incrementa el grado del polinomio tanto el error de entrenamiento como de interpolación decrecen; sin embargo, a partir del polinomio de grado seis el error de interpolación comienza a aumentar, y el de entrenamiento tiende a cero, lo cual es síntoma del sobreajuste (Bishop, 1995). Donde, el polinomio de grado nueve es el peor de los casos, este modelo responde perfectamente a los datos de entrenamiento, pero no tiene capacidad para generalizar nuevos datos.

Otro síntoma del sobreajuste es la alta fluctuación o varianza de la magnitud de los parámetros, punto crítico que se debe tratar de algún modo; en la Figura 2-6 se presenta un diagrama de Hinton, donde se observa que los pesos w4, w5 y w6 son en magnitud exageradamente grandes respecto a los demás, lo cual evidencia la saturación de los parámetros.

Figura 2-5: Comportamiento del RMSE de entrenamiento y de interpolación al aumentar el grado del polinomio sin regularizar.

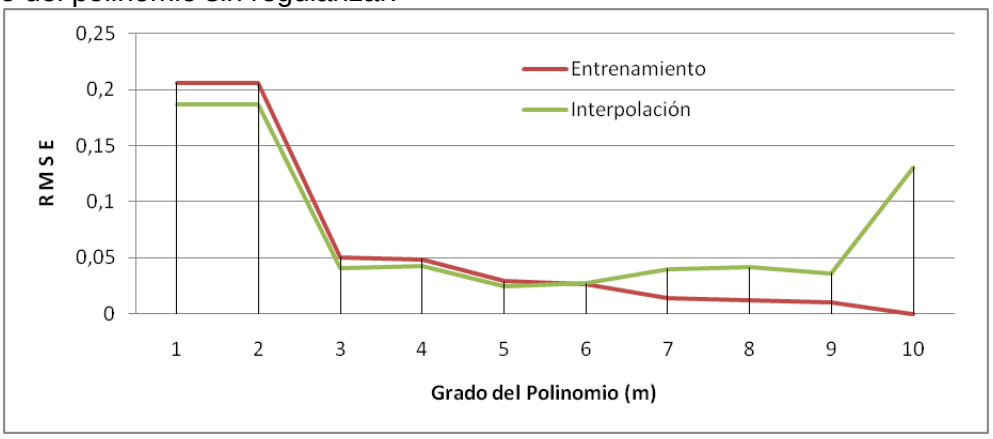
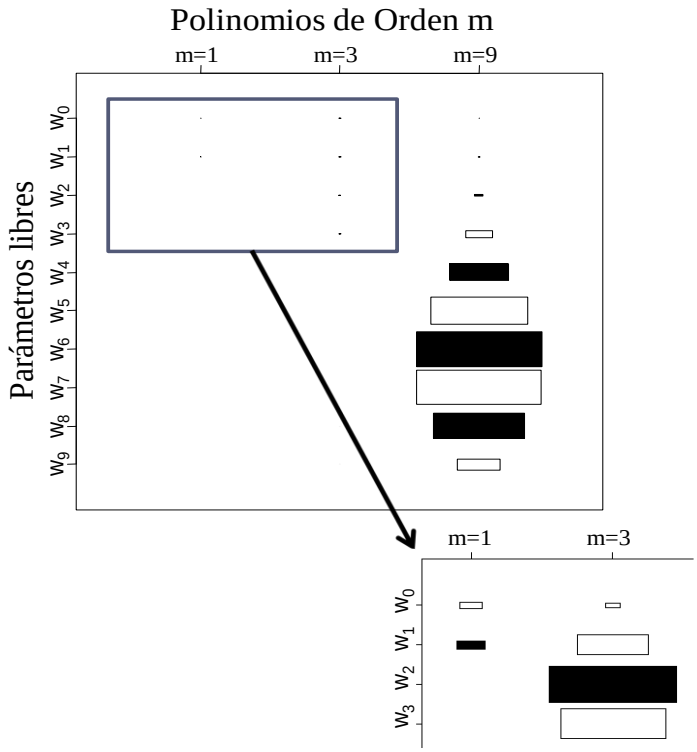


Figura 2-6: Diagrama de Hinton, de los pesos de los modelos de grado 1, 3 y 9



Esta serie de experimentos ha permitido evidenciar y analizar los síntomas del sobreajuste, mostrando la complejidad de este fenómeno en un caso sencillo como el ajuste de curvas con polinomios. De hecho uno de los interrogantes que se aborda en esta Tesis ¿En la práctica cómo se puede controlar el sobreajuste con las estrategias de regularización y cuáles problemas surgen para controlarlo efectivamente?

2.2 El control del sobreajuste mediante regularización

Una de las maneras más aceptadas en la literatura para controlar el problema de sobreajuste en las redes neuronales, son las estrategias de regularización de eliminación y descomposición de pesos. En esta sección se realizan una serie de experimentos con el fin de corroborar en la práctica el efecto de la estrategia de regularización de Descomposición de Pesos y de resaltar las dificultades que surgen al aplicar tal estrategia.

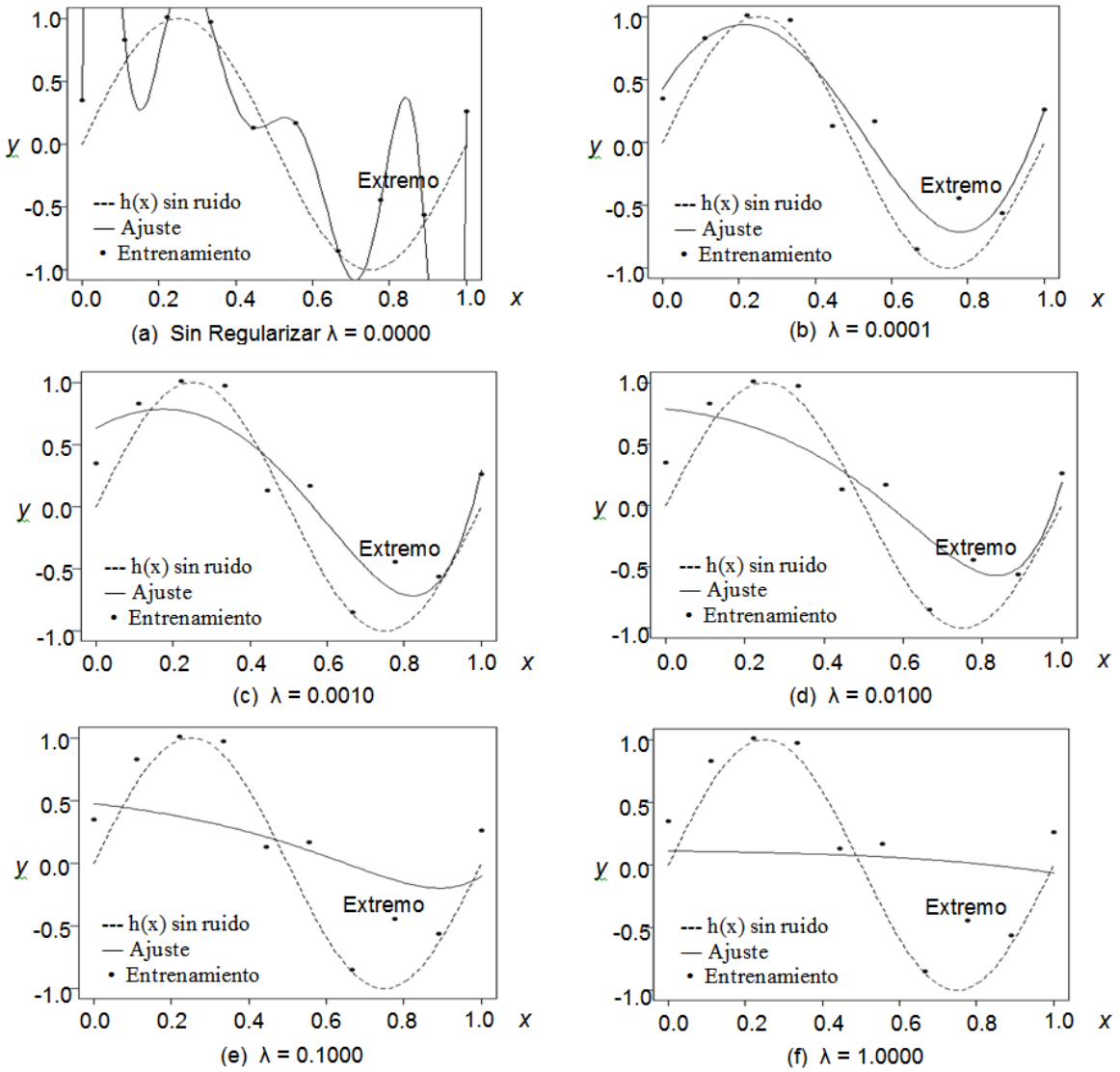
Asumiendo que la estrategia de descomposición de pesos es adecuada para regularizar un polinomio de grado m (ver Ecuación (2.2)), es necesario hallar un valor adecuado para λ. Para esto se regularizó el polinomio de grado nueve (el que padece de sobreajuste) con varios valores para λ; dado que su dominio es (0, +∞), empíricamente se consideró que la incidencia de la estrategia de regularización puede ser alta, dado que el polinomio de grado nueve está sobreparametrizado (se tienen 9 grados de libertad y 10 datos de entrenamiento); entonces, experimentalmente se regularizó con los valores de {0.5, 1, 2, 3, 5}, se observó que a partir de 1, la curva de ajuste se suavizó más de lo necesario.

Para este caso particular, valores mayores que 1 para λ suavizan más de lo necesario la curva de ajuste; entonces, se experimentó con varios valores de λ entre (0, 1], los más representativos son {0, 0.0001, 0.001, 0.01, 0.1, 1}, los resultados se resumen en la Figura 2-7, en ésta se evidencia que entre más tiende λ a cero, menos influencia tiene la estrategia de regularización, pero más mejora la generalización; por otro lado, entre más se tiende a 1, más se suaviza la curva de interpolación y empeora la generalización. Adicionalmente, se puede notar que los datos extremos, continúan teniendo una alta incidencia en el entrenamiento del modelo, a pesar de que se aplique la regularización. Mediante esta exploración, experimental se puede considerar que λ = 0.0001 es un valor

adecuado para regularizar (ver Figura 2-7 (b)), pero no se puede garantizar que sea el mejor.

Con el fin de ilustrar que el valor de λ no está necesariamente entre (0, 1], un experimento similar fue reportado por Bishop (1995), en el cual ajustó una red RBF para un conjunto de puntos dado por la Ecuación (2.2) y regularizó también con descomposición de pesos, encontró que el λ adecuado es 40; mientras que un valor de 1000 sobre-suaviza la curva de ajuste.

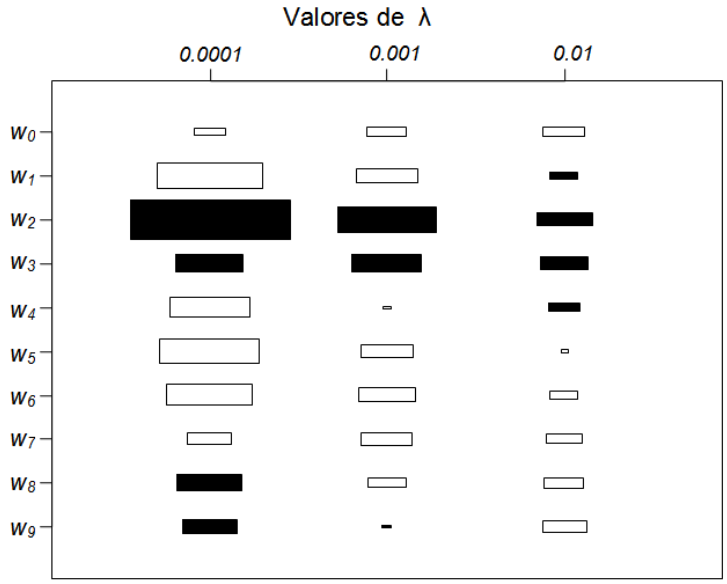
Figura 2-7: Regularización de un polinomio grado 9.



Aparentemente el problema se ha controlado con λ = 0.0001, pero aún no se ha analizado la magnitud de los pesos con el fin de comprobar si la estrategia de

regularización realmente funciona como se espera en la práctica. Para esto en la Figura 2-8, se presenta un diagrama de Hinton con la magnitudes de los pesos del polinomio de grado 9 regularizado con λ = {0.0001, 0.001, 0.01}. Para λ = 0.0001 (el valor considerado adecuado) se observan altas fluctuaciones de los pesos, es decir, aún se tienen pesos muy grandes en magnitud (w_1, w_2 y w_5) respecto a los demás; el comportamiento esperado es el que se obtiene con λ = 0.01 (considerado no adecuado) las fluctuaciones de las magnitudes de los pesos que no son de exceso son bajas; otro hecho preocupante con λ = 0.0001 es que existe una alta varianza de los pesos. Si se analizan los pesos obtenidos con valores de λ que se consideraron no adecuados, para 0.001 se observa que w_4 y w_9 son pesos de exceso; mientras que, para 0.01 el peso w_5.

Figura 2-8: Diagrama de Hinton de los pesos del polinomio de grado nueve, regularizado con Descomposición de Pesos y varios valores de λ.



Entonces, experimentalmente se encontró que λ = 0.0001 podría ser un valor adecuado para la incidencia de la regularización, dado este permite encontrar un comportamiento similar al deseado; sin embargo, el análisis de las magnitudes de los parámetros hallados con este valor de λ muestra que el comportamiento de estos no es el esperado según la teoría de regularización. Esto puede ser debido a varios factores: el valor encontrado de λ realmente no es el adecuado; la estrategia de regularización no es la apropiada para el modelo; la estrategia de regularización de descomposición de pesos no es tan robusta en la práctica como se afirma en la literatura.

2.3 El problema del sobreajuste en las redes Cascada-Correlación

Como ya se ha mencionado, una red CC puede adolecer de sobreajuste debido básicamente a dos causas: la primera está relacionada con la existencia de datos extremos (*outliers*) en el conjunto de entrada; la segunda con el tamaño óptimo de la red.

La primera causa se puede abordar mediante la regularización de las conexiones entre la capa oculta y la capa de salida, usando la estrategia de regresión de Borde (*Ridge Regression*) propuesta por Hoerl y Kennard (1970). La idea central de esta estrategia es controlar la varianza de los parámetros (pesos) buscando el equilibrio entre sesgo y varianza (*bias variance trade-off*); para más detalles sobre este aspecto, se sugiere ver los trabajos de Hoerl y Kennard (1970) y Marquardt y Snee (1975). Esta estrategia de regularización puede reducir la varianza de los pesos y minimizar el efecto de los datos extremos, y consecuentemente, reducir el error en validación; fue incorporada por Villa y Velásquez (2010) en las redes CC para el pronóstico de series de tiempo, obteniendo modelos con mejor capacidad de generalización que otras técnicas tradicionales como MLP y que el mismo tipo de redes sin regularizar.

Por otro lado, algunos autores también han utilizado regresión de Borde para regularizar redes, por ejemplo: Dutoit *et al.* (2009) utilizan regresión de Borde para regularizar redes ESN (*echo states network*) y muestran que ésta estrategia reduce el tamaño de las conexiones y no elimina ninguna de ellas, de tal manera que se reduce el error de validación (Dutoit, y otros, 2009).

Respecto a la segunda causa, que corresponde a cómo seleccionar u obtener el tamaño óptimo de una red neuronal, en la literatura se disponen de diversas técnicas para controlar el tamaño de la red, entre estas se tienen estrategias de regularización que, en general, siguen un enfoque de reducción o podado de la red *(Network Pruning)* (Palit & Popovic, 2005). La reducción consiste en comenzar con una red relativamente grande y sistemáticamente anular o reducir la importancia de algunas conexiones de acuerdo a algún un criterio definido, siendo posible eliminar neuronas, mientras se mejore la capacidad de generalización de la red; una de las más importantes críticas a este método

es que no se sabe si la red inicial es suficientemente grande como para que tenga neuronas innecesarias (Villa, Velásquez, & Souza, 2008). Este enfoque se usa preferiblemente cuando se desea diseñar redes que posean una gran capacidad de generalización, por ejemplo, para problemas como la predicción de series de tiempo o la clasificación de patrones, entre otros (Palit & Popovic, 2005).

En el enfoque de reducción se encuentran varias estrategias de regularización; las cuales, se utilizan en la metodología de regularización de Tikhonov (1973), entre otras: Descomposición de Pesos (*Weight Decay*) (Hinton, 1989); Eliminación de Pesos (*Weight Elimination*) (Weigend, Rumelhart, & Huberman, 1991); Olvido de Pesos (*Weight Forgetting*) (Ishikawa, 1989; Ishikawa, 1996) y Suavizado Aproximado (*Approximate Smoother*) (Moody & Rögnvaldsson, 1997);

El objetivo principal de estas estrategias, es construir una red con un tamaño óptimo, que sea menos propensa a aprender el ruido en los datos de entrenamiento y a incurrir en el sobreajuste, y por ende, puede generalizar con mayor precisión en un tiempo computacional menor que una red de mayor tamaño. Sin embargo, en la literatura más relevante no se especifica:

- Las razones metodológicas o teóricas para seleccionar una estrategia de regularización específica entre varias alternativas para los MLP para el pronóstico de series de tiempo, y mucho menos para las redes CC.
- Las implicaciones metodológicas, conceptuales y prácticas que acarrea la aplicación de cada una de las estrategias de regularización mencionadas, en las redes neuronales artificiales para el pronóstico de series de tiempo.

Tradicionalmente se han usado las estrategias de eliminación y descomposición de pesos para regularizar los pesos de las conexiones entre las neuronas de las redes MLP (Palit & Popovic, 2005; Chow & Cho, 2007); como también para las redes CC (Xu & Nakayama, 1997; Villa, Velásquez, & Souza, 2008). Estas dos estrategias fueron incorporadas en las redes CC por Villa *et al.* (2008) para el pronóstico de series de tiempo; los autores mostraron experimentalmente que la regularización permite controlar el sobreajuste y encontrar modelos con una alta capacidad de generalización.

Sin embargo, el problema no es solamente seleccionar una determinada estrategia de regularización, también es necesario determinar qué tanto debe incidir tal estrategia sobre el entrenamiento de la red; esto se controla a través de un factor de regularización. Éste factor es un número entre cero e infinito, si es muy bajo la regularización no tiene efecto y se pude continuar incurriendo en sobre entrenamiento, si por el contrario es muy alto la estrategia tiene mucho efecto y la red aprende poco.

En la literatura más relevante no se ha indicado una manera sistemática o metodológica para seleccionar éste factor para una red cascada correlación y siempre se deja a criterio experto, ver por ejemplo (Xu & Nakayama, 1997; Chow & Cho, 2007; Villa, Velásquez, & Souza, 2008); adicionalmente, no se ha estudiado a cabalidad cuál es el efecto de otros tipos de estrategias de regularización, diferentes a descomposición y eliminación de pesos, sobre el entrenamiento de redes CC.

2.4 Evaluación experimental de algunas técnicas de regularización para redes CC

La regularización puede solucionar el problema del sobreajuste en las redes neuronales; sin embargo, para que su aplicación sea efectiva es necesario tomar dos decisiones que son críticas y subjetivas: la primera es seleccionar la estrategia de regularización; y la segunda es determinar el parámetro de regularización (Evgeniou, Poggio, Pontil, & Verri, 2002; Hagiwara, 2002; Chow & Cho, 2007). En esta sección se abordará la primera, entonces se evalúa el comportamiento de diferentes términos de penalización compleja (estrategia de regularización) con el fin de verificar el comportamiento esperado teórico, mediante un análisis de los resultados de aplicar tales estrategias a las redes Cascada-Correlación para el pronóstico de algunas series de tiempo tradicionales; para finalmente recomendar cuál de las evaluadas es más adecuada para controlar el problema del sobreajuste en las redes CC.

Entonces, se presenta una comparación entre una red CC sin regularizar y varias redes CC regularizadas con las estrategias de eliminación y descomposición al pronosticar la serie de tiempo “Pasajeros de una Aerolínea” de Box y Jenkins (1976). Esta serie ha sido estudiada en la literatura por (Faraway & Chatfield, 1998) utilizando un MLP, por (Ghiassi, Saidane, & Zimbra, 2005) mediante DAN2. Además, para las correspondientes

estrategias de regularización se consideran los parámetros definidos en las Tabla 2-1 y Tabla 2-2 .

Tabla 2-1: Parámetros de regularización para el esquema de regularización de descomposición de pesos.

Descomposición de Pesos				
Parámetro	**CC-D_1**	**CC-D_2**	**CC-D_3**	**CC-D_4**
λ	0.001	0.010	0.050	0.100

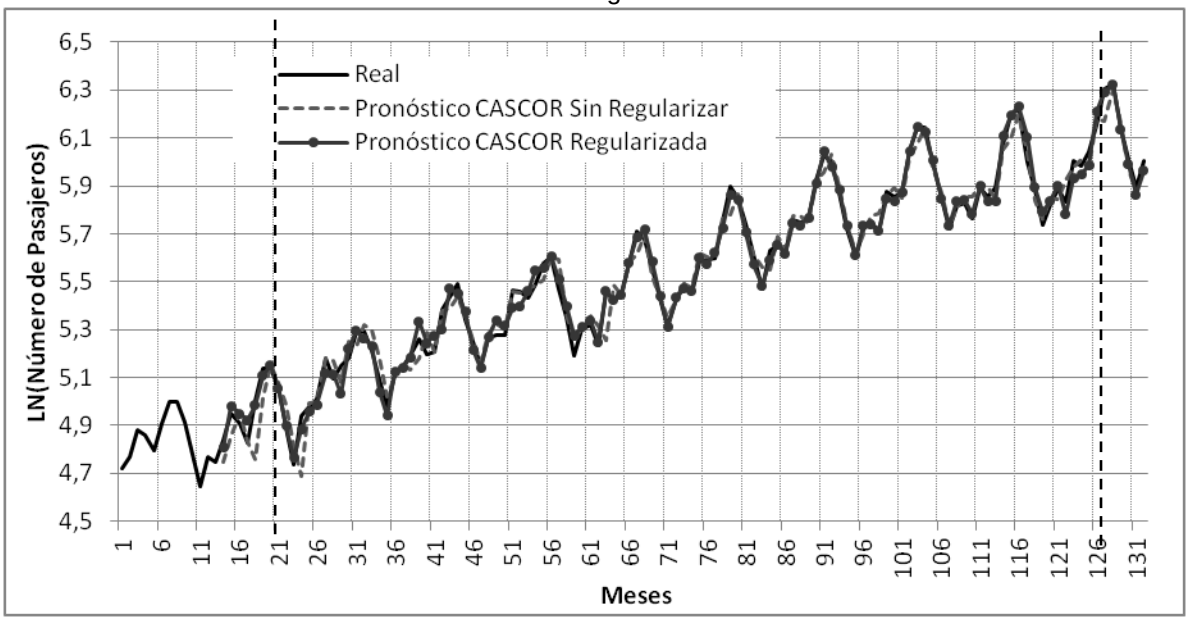
Tabla 2-2: Parámetros de regularización para el esquema de regularización de eliminación de pesos.

Descomposición de Pesos								
Parámetros	**CC-E_1**	**CC-E_2**	**CC-E_3**	**CC-E_4**	**CC-E_5**	**CC-E_6**	**CC-E_7**	**CC-E_8**
λ	0.001	0.010	0.050	0.100	0.001	0.010	0.050	0.100
w0	10	10	10	10	100	100	100	100

Para comparar la habilidad de las redes CC sin regularizar y regularizadas, se calcula la sumatoria del error medio cuadrático (SSE) de entrenamiento y validación, al pronosticar la serie de tiempo con 17 modelos de redes CC: sin regularizar (Tabla 2-3 y Tabla 2-4); regularizadas con descomposición denotados por CC-D_i (Tabla 2-3 yTabla 2-4); y eliminación de pesos detonado por CC-E_j (Tabla 2-5 y Tabla 2-6). Los índices *i* descomposición de pesos y *j* de eliminación, denotan una combinación específica de los parámetros de regularización, los cuales están dados en las Tabla 2-1 y Tabla 2-2, respectivamente. Además, los datos de la serie se transformaron utilizando la función logaritmo natural (*base - e*); para el pronóstico, se usaron los primeros 120 datos para entrenamiento y los 12 últimos para validación. En la Figura 2-9 se grafican los valores reales y pronosticados de la serie, usando el mejor modelo CC regularizado y el mejor sin regularizar. En la gráfica se observa que la red CC regularizada se ajusta mejor a la serie que la red sin regularizar.

En las Tabla 2-3 y Tabla 2-4 se resumen los resultados de entrenamiento y validación, respectivamente; al regularizar mediante descomposición de pesos (CC-D_i). Mientras que en las Tabla 2-5 y Tabla 2-6 se presentan los resultados al pronosticar con redes CC regularizadas con eliminación de pesos (CC-E_j). Para las Tabla 2-3 a Tabla 2-6, la columna CC indica que el pronóstico se realizó sin ninguna estrategia de regularización.

Figura 2-9: Valores real y pronosticado para la serie de pasajeros de una aerolínea, con un modelo CC Regularizado



Los resultados presentados en la Tabla 2-3 indican que al hacer λ=0.001 (columna CC-D_1), es indiferente utilizar el modelo 6 ó 7 para entrenamiento, dado que logran mismos errores. Asimismo, es indiferente usar los modelos: 8, 9 ó 10; 11 ó 12; y 13, 14, 15, 16 ó 17; son claramente 4 grupos de modelos. Al aumentar λ a 0.01 (CC-D_2), es indiferente utilizar en entrenamiento: 6 ó 7; 8, 9 ó 10; 11 ó 12; 13 ó 14; 15, 16 ó 17; son 5 grupos. Entonces, esta técnica de regularización, para este caso, permitiría reducir la cantidad de modelos. Haciendo λ= 0.05 (CC-D_3) se distinguen los mismos grupos de D_2 pero con un error mayor, igualmente cuando se aumenta λ a 0.1 (CC-D_4) también aumenta el error. Además, en la validación (Tabla 2-5), similar al entrenamiento en varios modelos el error obtenido de fue igual, en CC-D_1 se tienen 4 grupos de modelos con el mismo error, en CC-D_2 3 grupos, y en CC-D_3 y CC-D_4 4 grupos. Tanto en entrenamiento como en validación, se observa que la descomposición de pesos logra que los errores varíen menos entre modelos, esto posibilita agruparlos y que sea indiferente utilizar cualquier modelo de un grupo específico.

Tabla 2-3: Valores del error cuadrático en entrenamiento para diferentes modelos regularizados con la estrategia de descomposición de pesos, pronosticando la serie del caso de estudio.

Descomposición de Pesos – SSE de Entrenamiento							
Modelo	Rezagos	Neuronas	CC	CC-D_1	CC-D_2	CC-D_3	CC-D_4
1	1, 2, 13	4	0.826	0.612	1.256	1.420	1.459
2	1, 4, 8, 12	3	0.228	0.337	0.467	0.844	1.031
3	1, 4, 8, 12, 13	4	0.106	0.223	0.513	0.849	0.983
4	1, 4, 8, 10, 12, 13	3	0.171	0.221	0.491	0.816	0.957
5	1 – 4	2	1.171	1.123	1.487	2.062	2.277
6	1 – 13	2	0.145	0.214	0.451	0.821	1.057
7	1 – 13	4	0.174	0.214	0.451	0.821	1.057
8	1, 12	2	0.301	0.343	0.391	0.435	0.454
9	1, 12	4	0.286	0.343	0.391	0.435	0.454
10	1, 12	10	0.242	0.343	0.391	0.435	0.454
11	1, 2, 12	2	0.334	0.335	0.457	0.690	0.768
12	1, 2, 12	4	0.255	0.335	0.457	0.690	0.768
13	1, 2, 12, 13	2	0.185	0.223	0.502	0.783	0.863
14	1, 2, 12, 13	4	0.184	0.223	0.502	0.783	0.863
15	1, 12, 13	1	0.183	0.223	0.473	0.644	0.684
16	1, 12, 13	2	0.186	0.223	0.473	0.644	0.684
17	1, 12, 13	4	0.154	0.223	0.473	0.644	0.684

Entre tanto, en entrenamiento y validación, los modelos 1–5, donde varían la cantidad de neuronas y los rezagos, los errores con CC-D_1 son cercanos a los obtenidos con CC e incluso algunos son menores (en entrenamiento los modelos 1 y 5; y en validación 1, 2, 3 y 4). Sin embargo, cuando se aumenta λ (se hace que el término de regularización tenga más importancia en la red) los errores aumentan, tal es el caso de las columnas CC-D_2, CC-D_3, CC-D_4. En los modelos 6 y 7, se mantienen fijos los rezagos, y al aumentar las neuronas ocultas el error de entrenamiento no varía, pero si cambia en redes CC sin regularizar. Similarmente, en los modelos: 8, 9 y 10, al aumentar las neuronas ocultas, primero en dos unidades, y luego en seis, los errores no cambian; 13 y 14 las neuronas se incrementan en dos unidades y los errores permanecen estables; y 15, 16 y 17, de 15 a 16 se aumenta una neurona, luego de 16 a 17 dos unidades y ocurre lo mismo.

Tabla 2-4: Valores del error cuadrático en validación para los modelos de la Tabla 2-3 regularizados con la estrategia de descomposición de pesos, pronosticando la serie del caso de estudio.

Descomposición de Pesos – SSE de Validación							
Modelo	Rezagos	Neuronas	CC	CC-D_1	CC-D_2	CC-D_3	CC-D_4
1	1, 2, 13	4	0.196	0.139	0.148	0.164	0.169
2	1, 4, 8, 12	3	0.036	0.022	0.031	0.079	0.106
3	1, 4, 8, 12, 13	4	0.020	0.014	0.036	0.078	0.096
4	1, 4, 8, 10, 12, 13	3	0.014	0.013	0.031	0.067	0.085
5	1 – 4	2	0.140	0.162	0.174	0.239	0.266
6	1 – 13	2	0.059	0.016	0.028	0.068	0.102
7	1 – 13	4	0.013	0.016	0.028	0.068	0.102
8	1, 12	2	0.033	0.020	0.028	0.035	0.038
9	1, 12	4	0.019	0.020	0.028	0.035	0.038
10	1, 12	10	0.046	0.022	0.028	0.035	0.038
11	1, 2, 12	2	0.023	0.022	0.032	0.062	0.073
12	1, 2, 12	4	0.039	0.022	0.032	0.062	0.073
13	1, 2, 12, 13	2	0.012	0.014	0.036	0.071	0.082
14	1, 2, 12, 13	4	0.012	0.014	0.036	0.071	0.082
15	1, 12, 13	1	0.011	0.014	0.036	0.057	0.062
16	1, 12, 13	2	0.011	0.014	0.036	0.057	0.062
17	1, 12, 13	4	0.010	0.014	0.036	0.057	0.062

Consecuentemente, los resultados experimentales (entrenamiento y validación) al pronosticar la serie con redes CC regularizadas mediante descomposición de pesos muestran que: se logra un error estable a pesar de que se aumente la cantidad de neuronas ocultas en un modelo de red CC; con λ relativamente pequeño (λ=0.001) se pueden lograr errores menores que los obtenidos con redes CC sin regularizar; al aumentar λ los errores continúan siendo estables, pero aumentan.

En la Tabla 2-5 se presentan los resultados experimentales de entrenamiento al pronosticar la serie con los 17 modelos de la Tabla 2-3 regularizados con la estrategia de eliminación de pesos, variando el parámetro λ y w_0 como se indica en la Tabla 2-2. Los resultados revelan que dejando $w_0=10$ fijo el error de entrenamiento de la red CC no regularizada se disminuye al hacer λ=0.001 (columna CC-E_1), se reduce aún más cuando se aumenta λ a 0.01 (CC-E_2) en todos los modelos. Sin embargo, el error incrementa cuando se aumenta λ a 0.05 (CC-E_3), pero al incrementar λ a 0.1 (CC-E_4), el error disminuye respecto a CC-E_3, es decir, CC-E_4 < CC-E_3. Luego, cuando *w0* es aumentado a 100 y se mantienen fijo, se nota que algunos modelos tienden a un error específico

aunque se aumenten el número de neuronas, si λ=0.001 (CC-E_5) los modelos: 6 y 7 tienen un error de 0.188; 8, 9 y 10 de 0.341; 11 y 12 de 0.331; 13 y 14 de 0.197; y 15, 16 y 17 de 0.198. Igualmente, cuando λ=0.05 (CC-E_7) los modelos tienden al mismo error, pero mayor que el logrado con λ=0.001. Además, con λ=0.01 (CC-E_6) y λ=0.1 (CC-E_8) los errores obtenidos son menores que los logrados con redes CC sin regularizar.

Tabla 2-5: Valores del error cuadrático de entrenamiento para los modelos de la Tabla 2-3 regularizados con la estrategia de eliminación de pesos, pronosticando la serie del caso de estudio.

Eliminación de Pesos – SSE de Entrenamiento									
Modelo	CC	CC-E_1	CC-E_2	CC-E_3	CC-E_4	CC-E_5	CC-E_6	CC-E_7	CC-E_8
1	0.826	0.732	0.729	0.874	0.642	1.002	0.803	1.030	0.736
2	0.228	0.227	0.222	0.328	0.210	0.332	0.195	0.337	0.231
3	0.106	0.103	0.101	0.169	0.107	0.197	0.109	0.222	0.108
4	0.171	0.118	0.086	0.174	0.111	0.194	0.114	0.220	0.119
5	1.171	0.870	0.520	0.936	0.934	1.060	0.889	0.846	0.882
6	0.145	0.089	0.081	0.150	0.084	0.188	0.082	0.213	0.094
7	0.174	0.116	0.115	0.155	0.117	0.188	0.123	0.213	0.119
8	0.301	0.305	0.267	0.315	0.291	0.341	0.293	0.343	0.300
9	0.286	0.276	0.239	0.316	0.284	0.341	0.276	0.343	0.280
10	0.242	0.221	0.211	0.296	0.229	0.341	0.216	0.343	0.215
11	0.334	0.244	0.222	0.308	0.224	0.331	0.244	0.335	0.244
12	0.255	0.223	0.207	0.305	0.230	0.331	0.191	0.335	0.201
13	0.185	0.162	0.147	0.174	0.137	0.197	0.154	0.223	0.159
14	0.184	0.136	0.131	0.166	0.122	0.197	0.139	0.223	0.161
15	0.183	0.181	0.171	0.184	0.178	0.198	0.176	0.223	0.181
16	0.186	0.161	0.138	0.179	0.129	0.198	0.170	0.223	0.163
17	0.154	0.143	0.119	0.168	0.125	0.198	0.137	0.223	0.143

Los resultados en validación (Tabla 2-6) al pronosticar con el esquema de eliminación de pesos muestran que cuando w_0=10 los errores de las columnas CC-E_1, CC-E_2, CC-E_3, CC-E_4 son relativamente cercanos a los obtenidos con redes CC sin regularizar, e incluso algunos son menores; sin embargo, al aumentar el número de neuronas ocultas el error aumenta. Mientras que si w_0 se aumenta a 100, los errores de los modelos tienden a un error, aunque se aumente el número de neuronas, y en algunos casos es menor al de las redes CC sin regularizar.

Tabla 2-6: Valores del error cuadrático de validación para los modelos de la Tabla 2-3 regularizados con la estrategia de eliminación de pesos pronosticando la serie del caso de estudio.

Eliminación de Pesos – SSE de Validación									
Modelo	CC	CC-E_1	CC-E_2	CC-E_3	CC-E_4	CC-E_5	CC-E_6	CC-E_7	CC-E_8
1	0,196	0,141	0,179	0,189	0,053	0,145	0,236	0,139	0,368
2	0,036	0,038	0,035	0,025	0,052	0,023	0,045	0,022	0,038
3	0,020	0,024	0,018	0,017	0,026	0,014	0,017	0,014	0,027
4	0,014	0,042	0,024	0,017	0,035	0,015	0,035	0,015	0,041
5	0,140	0,156	0,101	0,164	0,158	0,140	0,148	0,140	0,152
6	0,059	0,014	0,014	0,016	0,011	0,015	0,013	0,016	0,026
7	0,013	0,023	0,018	0,017	0,012	0,015	0,013	0,016	0,026
8	0,033	0,026	0,030	0,022	0,019	0,020	0,030	0,020	0,051
9	0,019	0,037	0,037	0,023	0,029	0,020	0,030	0,020	0,051
10	0,046	0,050	0,050	0,028	0,059	0,020	0,030	0,020	0,051
11	0,023	0,037	0,040	0,027	0,036	0,023	0,046	0,022	0,037
12	0,039	0,049	0,075	0,028	0,042	0,023	0,046	0,022	0,037
13	0,012	0,019	0,019	0,016	0,020	0,014	0,026	0,014	0,016
14	0,012	0,024	0,022	0,018	0,022	0,014	0,026	0,014	0,016
15	0,011	0,015	0,015	0,013	0,014	0,013	0,023	0,014	0,015
16	0,011	0,025	0,016	0,014	0,016	0,013	0,023	0,014	0,015
17	0,010	0,027	0,029	0,016	0,026	0,013	0,023	0,014	0,015

2.5 Conclusiones

Los resultados experimentales (entrenamiento y validación) al pronosticar la serie con redes CC regularizadas mediante eliminación de pesos muestran que: cuando $w_0=100$ se logra un error estable a pesar de que se aumente la cantidad de neuronas ocultas en un modelo de red CC; y con diferentes combinaciones de λ y w_0, *e.g.* λ=0.01 y $w_0=100$, se pueden lograr errores menores que los obtenidos con redes CC sin regularizar.

Hasta este punto se ha comprobado experimentalmente los efectos de la regularización con descomposición y eliminación de pesos; según los resultados, estas técnicas han permitido encontrar modelos con mejor capacidad de predicción, siendo más beneficioso aplicar la estrategia de descomposición de pesos; sin embargo, ninguna de las dos estrategias muestra controlar efectivamente el sobreajuste, dado que no tienen en cuenta la varianza o desviación estándar de los datos ni la de los parámetros. Por tanto tales, estrategias no permiten seleccionar cuáles parámetros son de exceso o innecesario.

Consecuentemente es favorable regularizar mediante las técnicas utilizadas con el fin de controlar el sobreajuste; sin embargo, aún quedan aspectos por verificar, entre estos cómo seleccionar sistemáticamente el factor de regularización.

3. Capítulo 3: Una nueva estrategia sistemática de selección de modelos en redes CC

En el Capítulo anterior se comprobó experimentalmente los efectos de la regularización con descomposición y eliminación de pesos; según los resultados, estas técnicas han permitido encontrar modelos con mejor capacidad de predicción, siendo más beneficioso aplicar la estrategia de descomposición de pesos. En este Capítulo se describe conceptualmente y se evalúa experimentalmente una nueva estrategia sistemática para la selección del factor de regularización, lo que permitirá seleccionar cuál es el modelo con mejor capacidad de generalización, la cual se basa en un proceso regulable que dependiendo de si es necesario o no usar la regularización permite realizar la selección del modelo mientras se controla el sobreajuste.

3.1 Introducción

Asumiendo que la estrategia de regularización es la adecuada y tiene el efecto esperado, el parámetro λ representa el nivel de incidencia que ésta tendrá en el entrenamiento de la red. Si λ es cero, el aprendizaje se realiza sin la penalización; evidentemente, la estrategia seleccionada no tendrá incidencia sobre el aprendizaje y éste se realizará sin restricción alguna; mientras que, si λ tiende a infinito el aprendizaje sólo se basará en la penalización impuesta, suavizando más de lo necesario la curva del error, tendiendo los parámetros a cero y evitando que la red generalice adecuadamente los datos de entrenamiento (Bishop, 1994; Haykin, 1999; Chow & Cho, 2007). Cuando se selecciona un valor adecuado de λ, la penalización permitirá generalizar adecuadamente los datos de entrenamiento (Chow & Cho, 2007).

Entonces, es posible determinar experimentalmente un rango de valores para λ; sin embargo, tal búsqueda puede resultar infructuosa, dado que el rango del factor de penalización es (0, +∞), y las múltiples combinaciones entre las entradas a la red y sus neuronas ocultas hacen que su valor cambie. Generalmente, en la literatura se

recomienda seleccionarlo mediante algún tipo de esquema heurístico; con base en esto, debe ser posible encontrar una estrategia para seleccionar sistemáticamente el valor de λ, con el fin de mejorar la capacidad de generalización de redes neuronales (Leung & Chow, 1999). Así, el objetivo de este capítulo es plantear una estrategia sistemática para la especificación del valor del parámetro de regularización (λ) con la estrategia de regularización de descomposición de pesos.

3.2 Método

3.2.1 Consideraciones

Como se ha expuesto anteriormente, encontrar el parámetro de regularización λ es un problema complejo dado que generalmente se requiere de algún conocimiento previo del problema o de la solución para encontrarlo (Reginska, 1996). Para problemas mal condicionados no-lineales, la selección éste parámetro es más complicada debido a que la función de riesgo total dada por la Ecuación (1.10) puede ser de variación lenta lo que no permitiría detectar fácilmente los cambios, este aspecto se discute para problemas lineales por Hansen (1998).

En cuanto a la selección del parámetro de regularización, los aportes que se han realizado han sido en su mayoría desde el punto de vista de problemas inversos lineales, los cuales han sido extendidos a problemas inversos no-lineales (*e.g.,* similares al entrenamiento de una red neuronal). La selección se puede realizar a través de dos tipos de métodos:

- Basados en conocimiento *a priori* y supuestos sobre la norma del error. El más extendido de estos es el basado en Principio de Discrepancia, usualmente atribuido a Morozov (1966).
- Los que no requieren la norma del error, esta información se extrae de los datos de entrenamiento. Los más extendidos son validación-cruzada generalizada (Wahba, 1990) y el criterio de L-Curva sus detalles se describen en (Hansen, 1998).

En el contexto de las redes neuronales la información que se tiene *a priori* sobre el error es escasa o nula, esto dificulta la idealización de un método fiable para seleccionar el parámetro de regularización (Hansen, 1998). A esto se le suma la complejidad propia del mundo no-lineal. Entonces, el primer tipo de métodos no aplica para este contexto, mientras que el segundo tipo tiene los siguientes limitantes:

- A diferencia del aprendizaje tradicional explicado en el Capítulo 1, la validación cruzada generalizada requiere que el conjunto de datos sea divido en tres subconjuntos de entrenamiento, validación y prueba (*training, validation and test)*, para el entrenamiento se usa el conjunto de entrenamiento y validación con el fin de obtener una buena capacidad de generalización mediante parada temprana (Girosi, Jones, & Poggio, 1995)*;* mientras que el conjunto de prueba sólo se usa para medir el desempeño de la red ante datos desconocidos; para más detalles consultar Suykens *et al.* (2005). En la Figura 3-1 se muestra una representación de la división de una serie de tiempo, como se puede observar tomar la decisión de cómo dividir el conjunto de datos suele ser una tarea crítica, dado que si se divide equivocadamente se corre el riesgo de eliminar información valiosa para el entrenamiento de la red. A esto se le suma otro aspecto crítico, λ es, a menudo, dependiente de los datos de entrenamiento (Chow & Cho, 2007). Entonces, este tipo de métodos, no es adecuado cuando el conjunto de datos es limitado, lo que ocurre la mayoría de las veces; además, agrega más complejidad al problema de seleccionar el parámetro de regularización, incorporando otra decisión crítica de cómo dividir el conjunto de datos.
- Respecto al criterio de L-Curva, es un método principalmente gráfico donde se presenta en escala logarítmica la norma de la solución normalizado *versus* la norma de la solución sin normalizar. Para más detalles se recomienda consultar Hansen (1998) y Duda *et al.* (2000). Aunque este método permite encontrar valores adecuados para el parámetro de regularización, requiere para cada cambio en la red, calcular la correspondiente L-Curva, lo cual es completamente ineficiente dado que cada vez que se agregue una neurona en la capa oculta o en la de entrada se requerirá conocer la correspondiente L-Curva, esta cantidad de cambios pueden ocurrir una cantidad indeterminada de veces en el aprendizaje de una red CC.

Figura 3-1: Partición del conjunto de datos en validación cruzada

Dadas las limitaciones que presentan los métodos tradicionales para determinar el parámetro de regularización, diversos autores coinciden en que la selección de λ adecuado para controlar el sobreajuste es una tarea compleja, se determina por criterio experto y es ajustado de tal manera que la regularización tenga más o menos impacto sobre la función objetivo (Xu & Nakayama, 1997; Setiono, 1997; Duda, Hart, & Stork, 2000; Palit & Popovic, 2005; Leung, Wang, & Sum, 2010).

Entonces, en la siguiente sección se propone un proceso ordenado y estándar que se ajusta al aprendizaje constructivo de la arquitectura de la red CC que incorpora la regularización para obtener modelos con adecuada capacidad de generalización.

3.2.2 La estrategia sistemática

Partiendo de la idea del aprendizaje constructivo de las redes CC, se propone un aprendizaje constructivo más avanzado que incorpora la selección de λ durante la construcción de la red, lo que a su vez permitirá determinar cuál es el modelo adecuado del conjunto de modelos evaluados que tenga la mejor capacidad de generalización. La estrategia que se propone se define a través de un proceso ordenado que incorpora un conjunto de pasos y de reglas que permiten seleccionar el mejor modelo en cada momento.

La estrategia sistemática se define en el Algoritmo 1, para esta se debe considerar los principios de que el error de entrenamiento disminuye a medida que se agregan neuronas en la capa oculta, se conoce la cantidad correcta de neuronas en la capa de entrada, el error de entrenamiento disminuye al incrementar el valor de λ. Cuando el error de entrenamiento aumenta es que no se requiere regularizar más, por consiguiente, no

es necesario aumentar λ. Finalmente, si el error de entrenamiento de una red sin regularizar es menor que el de una red regularizada, evidentemente no se requiere regularizar.

Algoritmo 1: Estrategia Sistemática para Seleccionar el Parámetro de Regularización

1. Iniciar el entrenamiento de una red CC con una cantidad fija de neuronas en la capa de entrada, igual al conjunto de datos de entrenamiento.
2. Entrenar una red CC sin neuronas en la capa oculta y Almacenar como **red base**.
3. Agregar una neurona en la capa oculta de la **red base** y Almacenar como la **red ampliada** conservando los parámetros calculados para **red base**.
4. *Entrenar la **red ampliada** con λ = 0 y Almacenar como **red actual**.*
5. *Incrementar λ, λ > 0.*
6. *Entrenar la **red ampliada** con λ y Almacenar como **red regularizada.***
7. *Si **red regularizada** logra un menor error de entrenamiento que la **red actual**, entonces:*
   7.1. La **red actual** es igual a la red regularizada, luego ir al paso 5.
   7.2. De lo contrario, ir al paso 8.
8. Si **red actual** logra un menor error de entrenamiento que la **red base**, entonces:
   8.1. La **red base** es igual a la **red actual.**
   8.2. Si se cumple criterio de finalización del entrenamiento ir al paso 9.
   8.3. Ir al paso 3
   8.4. De lo contrario, ir al paso 9.
9. Retornar **red base.**
10. Fin del Entrenamiento.

En general, la base de la estrategia son los pasos 1, 2, 3 y 8 correspondientes a la arquitectura de red CC, mientras que los pasos 4, 5, 6 y 7 son la propuesta que se incorpora en tal arquitectura para hallar dinámicamente el valor de λ. En general, los criterios de finalización del entrenamiento son: cuando se alcance una cantidad máxima de neuronas en la capa de oculta (se supera la cantidad de datos disponibles), o se alcance un delta del error deseado; y la estrategia de regularización será la de descomposición de pesos.

A continuación, se pondrá a prueba la estrategia planteada, al pronosticar la serie de Linces Canadienses.

3.3 Desempeño de la estrategia

En ésta sección se evidencia experimentalmente las bondades de la estrategia sistemática para hallar un valor adecuado para λ usando la estrategia de descomposición de pesos, para pronosticar la serie de tiempo de Linces Canadienses, ampliamente usada en la literatura; en esta serie se encuentra registrada la cantidad de linces capturados anualmente, desde 1821 hasta 1934, en los alrededores del río Mackenzie ubicado en el distrito de Northem, Canadá. Esta serie fue estudiada por (Campbell & Walker, 1977), (Rao & Gabr, 1984), y (Zhang G. , 2003). Los datos de la serie se transformaron utilizando la función logaritmo base 10; de sus 114 datos se tomaron los 100 primeros para entrenamiento y los últimos 14 para validación tal como se ha realizado en estudios pasados.

Para comenzar se pronostica la serie sin utilizar las estrategias de regularización, en la Figura 3-2 se presenta el comportamiento del error, tanto en entrenamiento como validación, en la cual se evidencia que cada vez que se agrega una neurona en la capa oculta el error de entrenamiento disminuye, pero a partir de la tercera el error de validación aumenta, lo que evidencia que la red está sobreajustando los datos. Se ha evidenciado que a partir de la tercera neurona oculta que se agrega, la red CC comienza a adolecer de sobreajuste. Luego, con el fin de evidenciar las bondades de la aproximación propuesta, se aplica la estrategia planteada, incrementando el valor de λ conforme a los siguientes valores {0, 0.0001, 0.001, 0.01, 0.1, 1}.

Los resultados del entrenamiento con la estrategia propuesta se resumen en Tabla 3-1, en estos se observa que al agregar la primera neurona oculta, el valor de cero para **λ** es posiblemente el más adecuado, dado que al aplicar la regularización con **λ=0.0001** el error alcanzado es mayor que sin regularizar; análogamente, cuando se agrega una segunda neurona, lo apropiado es no regularizar, es decir, **λ = 0**; a partir de la neurona 3 con **λ=0.01** el error disminuye más que sin regularizar y que con **λ = {0,.0001, 0,001}**. NC son errores que no fue necesario calcular, dado que según la estrategia no es necesario evaluar.

Figura 3-2: Error de entrenamiento y validación al pronosticar la serie de tiempo de Linces Canadienses con redes CC sin regularizar, aumentando la cantidad de neuronas en la capa oculta.

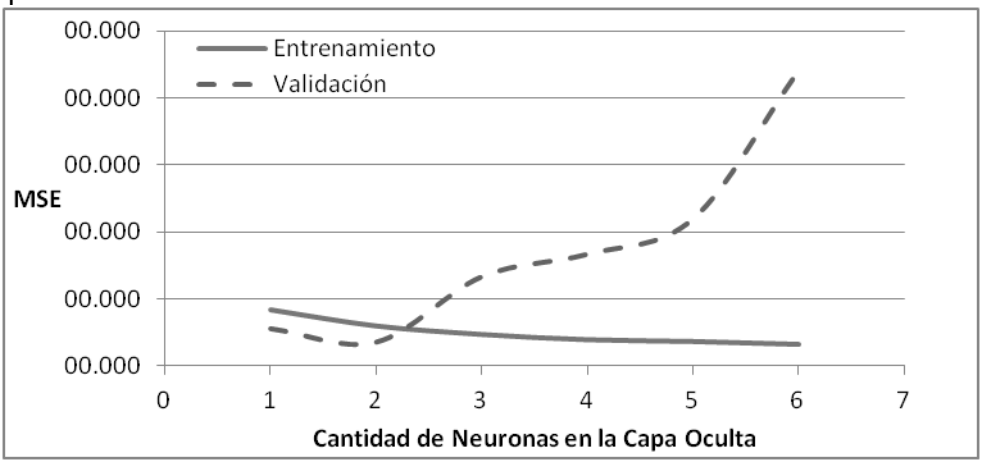


Tabla 3-1: Error cuadrático medio de entrenamiento al modelar la serie de tiempo de Linces Canadienses con redes CC variando **λ** y aumentando neuronas en la capa oculta. NC: No Calculado. H: Neuronas Ocultas

	Factor de regularización (λ)					
H	**0**	**0.0001**	**0.001**	**0.01**	**0.1**	**1**
1	***0.0418***	0.0443	NC	NC	NC	NC
2	***0.0300***	0.0310	NC	NC	NC	NC
3	0.0238	0.0212	0.0203	***0.0193***	18.6053	NC
4	0.0200	0.0199	0.0199	***0.0170***	17.5540	NC
5	0.0186	0.0185	0.0184	***0.0116***	19.1241	NC
6	0.0165	0.0158	0.0134	***0.0115***	19.1234	NC

Ahora se requiere evaluar qué tan buena es la capacidad de generalización que se puede lograr con la estrategia planteada; para esto, en la Tabla 3-2, se resumen los errores de validación correspondientes a la Tabla 3-1. Cuando no se regulariza, es decir **λ = 0**, el error de validación aumenta a partir de la tercera neurona mientras que el de entrenamiento disminuye, lo cual evidencia el sobre entrenamiento; al regularizar con **λ = 0.01** el error disminuye, controlando el sobre entrenamiento. Tal bondad se evidencia claramente en la Figura 3-3, donde se muestran los mejores errores obtenidos con la estrategia propuesta, se observa que tanto el error de entrenamiento como de validación disminuyen.

Figura 3-3: Error de entrenamiento y validación al pronosticar la serie de tiempo de Linces Canadienses con redes CC regularizadas, aumentando la cantidad de neuronas en la capa oculta.

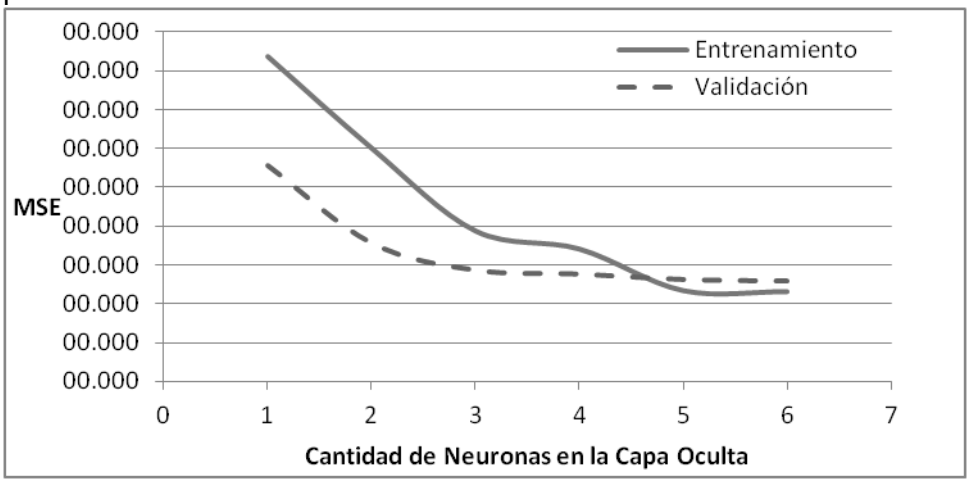


Tabla 3-2: Error cuadrático medio de validación al pronosticar la serie de tiempo de Linces Canadienses con redes CC variando **λ** y aumentando neuronas en la capa oculta. NC: No Calculado.

	Factor de regularización (λ)					
H	**0**	**0.0001**	**0.001**	**0.01**	**0.1**	**1**
1	**0.02788**	0.02901	NC	NC	NC	NC
2	**0.01789**	0.02731	NC	NC	NC	NC
3	0.06666	0.02511	0.02101	**0.0144**	19.423	NC
4	0.08335	0.02453	0.02067	**0.0139**	19.124	NC
5	0.10979	0.02337	0.02012	**0.0132**	18.451	NC
6	0.22022	0.02321	0.01982	**0.0130**	17.603	NC

3.4 Conclusiones

La estrategia propuesta muestra ser una opción adecuada para seleccionar sistemáticamente el factor de regularización y consiguientemente permite obtener el mejor modelo de la familia de modelos evaluados; además, es sofisticado por que en cada iteración aprovecha la información generada en iteraciones anteriores, sin necesidad de reprocesar búsquedas; adicionalmente, por basarse en los principios mencionados se puede evidenciar que converge. Entonces, sólo resta proponer una estrategia de regularización que garantice el control de la magnitud de los pesos, lo cual se discutirá en el siguiente capítulo.

4. Capítulo 4. La varianza y la desviación estándar como método de regularización.

4.1 Introducción

A lo largo de Capítulos anteriores se ha evidenciado la efectividad y eficacia de las estrategias de regularización y de la estrategia sistemática de selección del parámetro de regularización para controlar el sobreajuste; además, se han resaltado las ventajas y retos que debe cumplir las estrategias de regularización vigentes. En este orden de ideas, se requiere de una estrategia para controlar efectivamente la magnitud de los pesos o parámetros de la red neuronal, dado que ya se evidenció con el ejemplo de los polinomios que las estrategias de regularización aún tienen falencias.

El decaimiento de pesos no regula efectivamente la magnitud de los parámetros, dado que es básicamente la norma de los pesos y no evalúa de manera precisa la dispersión de los parámetros; entonces, con el fin de controlar masiva y efectivamente la magnitud de los pesos en esta Tesis se propone usar la varianza y la desviación estándar como estrategias de regularización para controlar el sobreajuste; dado que estas medias de dispersión o variabilidad, permiten calcular efectivamente la dispersión de una distribución, indicando por medio de un valor si los diferentes valores de los parámetros están muy alejados de la media. Cuanto menor sea este valor, menor será la variabilidad; mientras que, entre mayor sea, más heterogénea será a la media, es decir, de este modo se puede evaluar si los pesos son similares o varían mucho entre ellos.

Entonces, el objetivo de este capítulo es proponer conceptualmente y validar experimentalmente la varianza y la desviación estándar como estrategias de regularización; además, incorporarlas en la estrategia sistemática propuesta en el capítulo anterior. Con el fin de cumplir el objetivo, éste capítulo está organizado como sigue: en la siguiente sección se describe cómo se incorpora la varianza y la desviación estándar como estrategias de regularización, seguidamente se evalúa

experimentalmente las nuevas estrategias incorporadas a la estrategia sistemática y finalmente se concluye.

4.2 Método

El principal objetivo de la regularización es hallar una solución estable con pesos o parámetros homogéneos usando algún tipo de función de penalización en la función objetivo; tal función debe penalizar adecuadamente la función objetivo cada vez que el algoritmo de entrenamiento se desvía del objetivo. Entonces, se propone usar la varianza y la desviación estándar de los parámetros como dos nuevas estrategias de regularización; si alguna de estas es relativamente alta, penalizará fuertemente a la función objetivo, obligando a suavizar la función de error. Entonces se tendrán dos nuevas funciones de riesgo total, con base en la Ecuación (1.10), la varianza como término de penalización compleja en la función de riesgo total:

$$R(W) = \xi_s(\mathrm{W}) + \lambda\xi_c(\mathrm{W}) = \sum_{t=1}^{T}(\hat{y}_t - y_t)^2 + \lambda\left(\frac{1}{N-1}\right)\sum_{h=1}^{H}\sum_{p=1}^{P}\left(w_{p,h} - \overline{w}\right) \tag{4.1}$$

Ahora, la desviación estándar como término de penalización compleja:

$$R(W) = \xi_s(\mathrm{W}) + \lambda\xi_c(\mathrm{W}) = \sum_{t=1}^{T}(\hat{y}_t - y_t)^2 + \lambda\sqrt{\left(\frac{1}{N-1}\right)\sum_{h=1}^{H}\sum_{p=1}^{P}\left(w_{p,h} - \overline{w}\right)} \tag{4.2}$$

Para las Ecuaciones (4.1) y (4.2), N es la cantidad de parámetros de la red dado por $H(P+2)+1$, mientras que $\overline{w}$ está dado por:

$$\overline{w} = \left(\frac{1}{N}\right)\sum_{h=1}^{H}\sum_{p=1}^{P}\left(w_{p,h}\right) \tag{4.3}$$

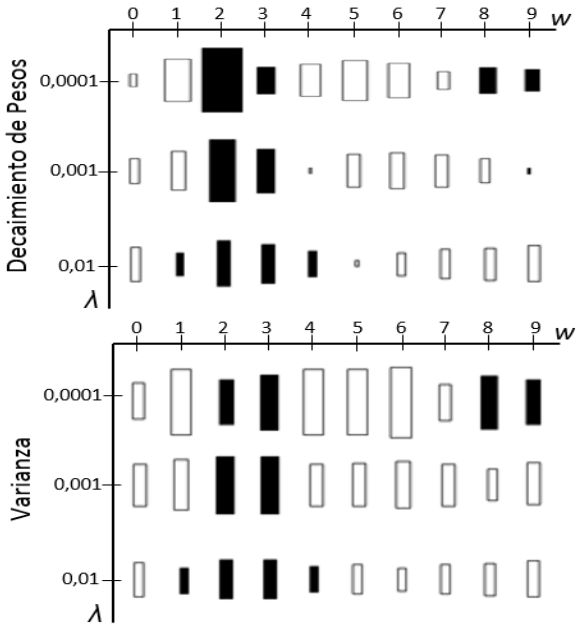
Como se puede observar en las Ecuaciones (4.1), (4.2) y (4.3), estos métodos también tiene en cuenta la cantidad de parámetros del modelo, a diferencia de decaimiento y eliminación de pesos. Respecto al parámetro de regularización, cada una de estas

estrategias se incorpora en la estrategia definida en el Capítulo 3, para conformar un Criterio de Selección de Modelo de red neuronal Cascada-Correlación.

Ahora, con el fin de evidenciar la efectividad del Criterio propuesto se retoma el experimento de la Sección 2.2, en el cual regularizar con λ = 0.0001 permite controlar el sobreajuste y reproduce adecuadamente el comportamiento de la función dada por la Ecuación 2.1, ver Figura 2-7 (b). Sin embargo, la técnica de descomposición de pesos no permite controlar efectivamente la magnitud de los parámetros tal como se evidencia en la Figura 4-1 (superior), donde para λ = 0.0001 se tienen pesos excesivamente grandes y otros pequeños, permitiendo que tengan una alta fluctuación.

La técnica que se ha propuesto, específicamente regularizar con la varianza, ha permitido controlar efectivamente el sobreajuste, dado que con λ = 0.0001 se ha obtenido el comportamiento esperado y además se ha controlado la magnitud de los pesos; esto se puede observar en la Figura 4-1 (inferior), donde los pesos con el factor de regularización hallado tienen poca fluctuación, y tienden a ser uniformes. Entonces, se ha evidenciado que regularizar con la varianza permite controlar efectivamente el problema del sobreajuste.

Figura 4-1. Diagrama de Hinton de los pesos del polinomio de grado nueve, regularizado con Descomposición de Pesos (superior) y con la Varianza (inferior) para varios valores de λ.



4.3 Desempeño del Criterio de Selección

Para demostrar la efectividad del Criterio de Selección, y por ende de los métodos de regularización propuestos integrados en la estrategia del Capítulo 3, se toma como guía la metodología seguida por Ghiassi (2005) y se usa el mismo conjuntos de datos de Makridakis (1998); entonces, en esta sección se presenta la evaluación experimental de los métodos de regularización de la varianza y desviación estándar incorporados en la estrategia del Capítulo 3 aplicándolos a dos tipos de problemas de series de tiempo.

El primer tipo de problema se refiere a tres series de tiempo usadas extensamente en la literatura: 'Pasajeros de una Aerolínea' de Box y Jenkins, 'Linces Canadienses' y el 'Número de Manchas Solares' de Wolf. El segundo tipo, corresponde a series de tiempo con comportamiento conocido: una estacional, 'las ventas de papel de impresión y escritura'; una no estacional en varianza, 'Envíos de equipos de contaminación'; y una no estacional en el proceso, 'Usuarios autenticados en un servidor de Internet'.

Cada serie se pronostica con diferentes modelos de redes CC, tanto sin regularizar como regularizadas con las nuevos métodos propuestos, los modelos CC regularizados con descomposición de pesos se identificarán por CC-DP, regularizados con la varianza por CC-VA y los regularizados con la desviación estándar con CC-DE; estos resultados se comparan con los reportados por (Ortíz, Villa, & Velásquez, 2007) y (Ghiassi, Saidane, & Zimbra, 2005) al pronosticar las series mencionadas con MLP y DAN2, respectivamente. Las redes CC fueron entrenadas con iRprop+.

4.3.1 Primer caso: Pasajeros de una aerolínea

Esta serie de tiempo contiene el registro del número total de pasajeros transportados por mes por una aerolínea, desde enero de 1949 hasta diciembre de 1960. Cada uno de los modelos presentados en la Tabla 4-1, fue estudiado por (Faraway & Chatfield, 1998). Para cada modelo, los datos de la serie se transformaron utilizando la función logaritmo natural (base - e); se usaron los primeros 120 datos para entrenamiento y los 12 últimos para validación, tal como fue realizado en (Faraway & Chatfield, 1998).

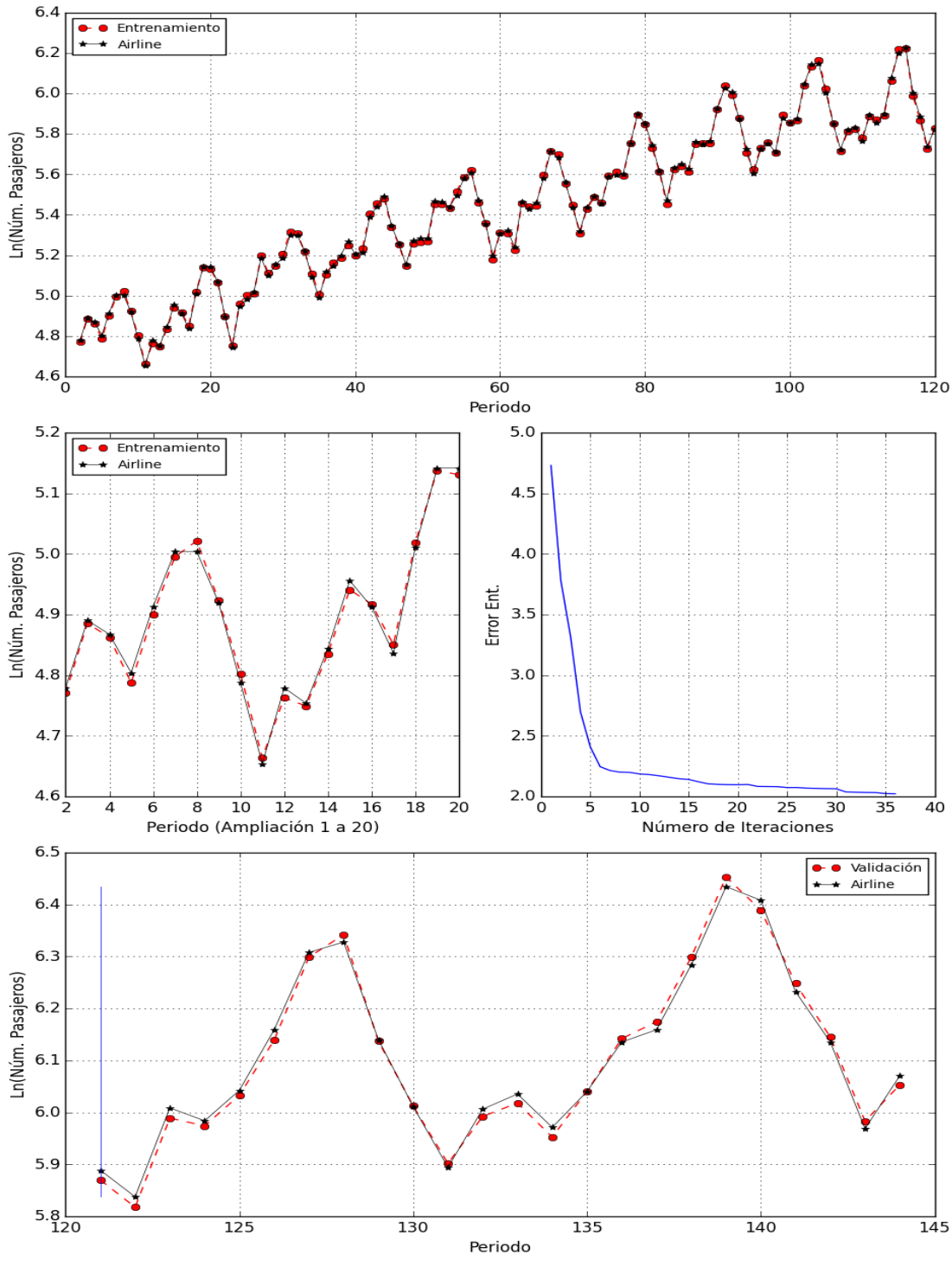
Faraway y Chatfield (1998) y Ghiassi *et al.* (2005) reportaron para cada uno de los modelos de la Tabla 4-1, la sumatoria del error cuadrático (SSE) para las muestras de entrenamiento y validación al predecir la serie. Igualmente para cada uno de dichos modelos, se procedió a realizar el pronóstico de la serie con los modelos respectivos de MLP, CC, CC-DP, CC-VA y CC-DE y se estimó el estadístico de ajuste para las muestras en entrenamiento y de validación. Se puede apreciar que para los modelos uno al cinco, excepto el tres, la regularización con la varianza (CC-VA) y con la desviación estándar (CC-DE) logró mejores resultados tanto en entrenamiento como validación, es decir, se encontraron modelos con mejor capacidad de generalización que los demás y los reportados con DAN2. Claramente, para ésta serie en particular los modelos de redes CC-VA y CC-DE fueron superiores que los MLP, ANN y DAN2.

Tabla 4-1: Valores del SSE para diferentes modelos pronosticando la serie del primer caso.

Núm. Mod.	Rezagos	SSE Entrenamiento							SSE Validación						
		ANN	MLP	DAN2	CC	CC-DP	CC-VA	CC-DE	ANN	MLP	DAN2	CC	CC-DP	CC-VA	CC-DE
1	1 – 13	0.26	0.20	0.17	0.19	0.18	**0.16**	**0.16**	1.12	0.87	0.23	0.27	0.21	**0.21**	**0.21**
2	1, 12	1.77	1.21	0.85	0.98	0.80	**0.75**	**0.70**	0.59	0.40	0.26	0.31	0.37	**0.24**	**0.21**
3	1, 2, 12	1.91	1.50	**0.44**	0.51	0.45	0.45	0.47	1.03	0.89	0.19	**0.25**	0.20	0.21	0.22
4	1, 2, 12, 13	0.81	0.77	0.30	0.39	0.29	**0.37**	**0.36**	0.52	0.51	0.29	0.31	0.21	**0.14**	**0.16**
5	1, 12, 13	0.84	0.72	0.24	0.45	0.25	**0.13**	**0.15**	0.62	0.58	0.22	0.33	0.19	**0.15**	**0.14**
6	1,2,13	–	0.76	0.12	0.15	0.17	**0.12**	**0.12**	–	0.45	0.30	0.35	0.33	**0.28**	**0.27**
7	1,4,8,12	–	0.81	0.24	0.27	0.25	**0.24**	**0.23**	–	0.54	0.25	0.33	0.27	**0.21**	**0.10**
8	1,4,8,12,13	–	0.54	0.19	0.15	0.17	**0.14**	**0.15**	–	1.32	0.29	1.20	0.28	**0.21**	**0.10**
9	1,4,8,10,12,13	–	0.39	0.11	0.10	0.14	**0.10**	**0.10**	–	1.98	0.11	1.48	0.23	**0.21**	**0.10**

En la Figura 4-2 se muestran los valores reales y los pronosticados usando el modelo 9 de CC-DE de la Tabla 4-1. Se puede apreciar que el modelo de red CC-DE se ajusta adecuadamente a la serie, tanto en entrenamiento como en validación reproduce el comportamiento de los datos.

Figura 4-2: Predicción con una red CC regularizada con la desviación estándar para la serie de pasajeros de una aerolínea. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



4.3.2 Segundo caso: Linces Canadienses

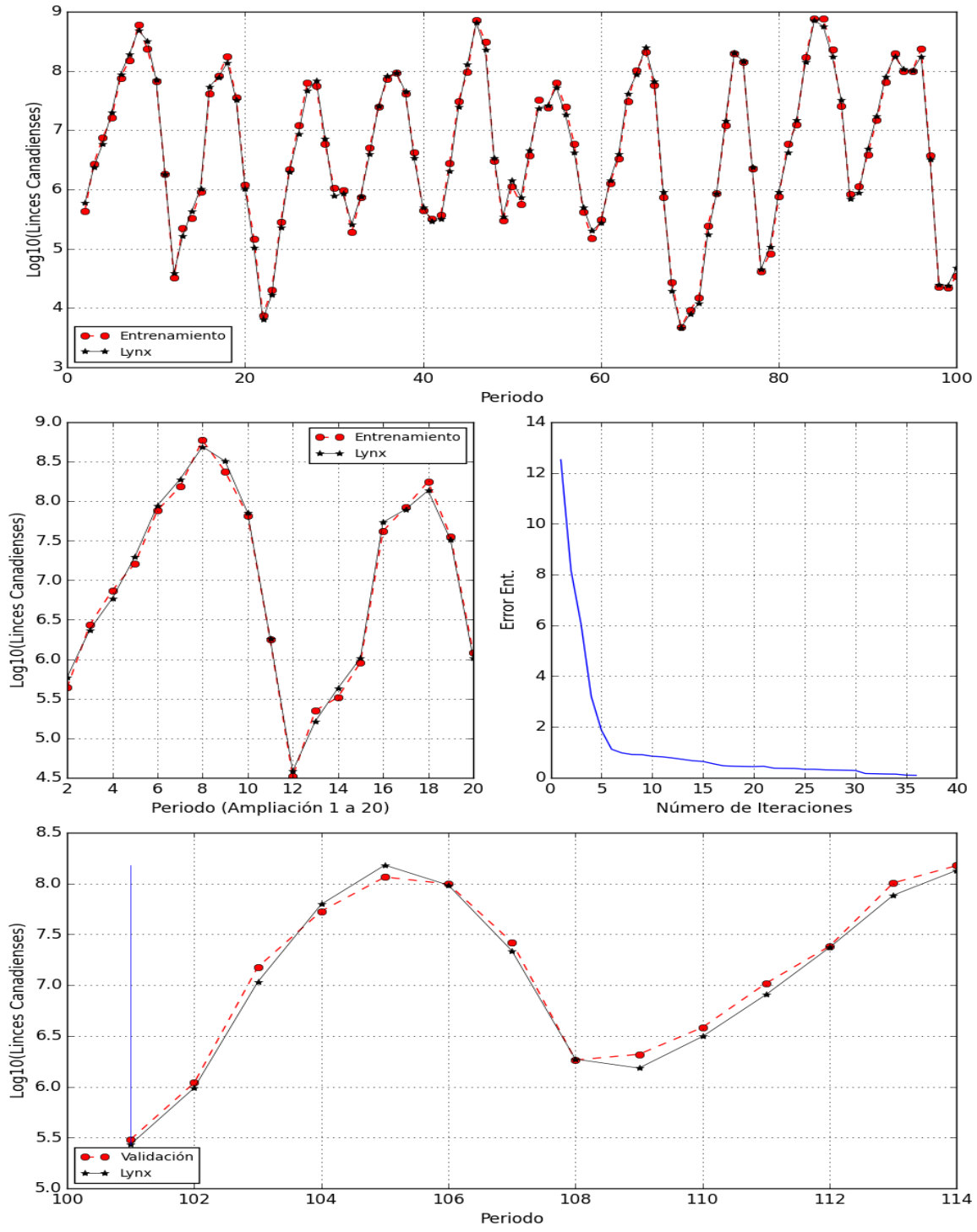
En esta serie se encuentra registrada la cantidad de linces capturados anualmente, desde 1821 hasta 1934, en los alrededores del río Mackenzie ubicado en el distrito de *Northem*, Canadá. Esta serie fue estudiada por (Campbell & Walker, 1977), (Rao & Gabr, 1984), y (Zhang G. , 2003). Los datos de la serie se transformaron utilizando la función logaritmo base 10; de sus 114 datos se tomaron los 100 primeros para entrenamiento y los últimos 14 para validación tal como se ha realizado en estudios pasados.

En la Tabla 4-2 se presenta el error cuadrático medio (MSE) calculado para distintos modelos estimados, se aprecia que similar al caso anterior, tanto en entrenamiento como en validación, con redes CC-VA y CC-DE se obtuvieron modelos con mejor capacidad de generalización que los demás, y se alcanzó el mismo nivel de generalización que el mejor modelo de DAN2 (ver modelo 6). Además, en los modelos 3 al 6, se evidencia que tanto el error de entrenamiento como de validación, no varían significativamente a pesar de que se agreguen más rezagos. En la Figura 4-3 se presentan los valores reales de la serie de tiempo y los pronosticados con el modelo 6 de CC-VA de la Tabla 4-2, en ésta gráfica se observa que la red CC-VA se ajusta adecuadamente a la serie, tanto en entrenamiento como en validación.

Tabla 4-2: Valores del error cuadrático medio para diferentes modelos pronosticando la serie del segundo caso.

Núm. Mod.	Rezagos	SSE Entrenamiento						SSE Validación					
		MLP	DAN2	CC	CC-DP	CC-VA	CC-DE	MLP	DAN2	CC	CC-DP	CC-VA	CC-DE
1	1,2,3,8,9,10	0.025	**0.001**	0.012	0.010	**0.001**	**0.001**	0.215	**0.008**	0.020	0.015	**0.008**	**0.008**
2	1 – 6	0.030	**0.001**	0.021	0.012	0.003	0.004	0.325	0.011	0.045	0.016	**0.010**	0.011
3	1,2,3,4,8,9,10	0.045	0.013	0.021	0.016	**0.010**	**0.010**	0.458	0.013	0.048	0.014	**0.009**	**0.009**
4	1 – 8	0.125	0.013	0.032	0.015	**0.010**	**0.010**	0.125	0.013	0.056	0.019	**0.009**	**0.009**
5	1 – 9	0.097	0.015	0.020	0.018	**0.010**	**0.010**	0.458	0.015	0.057	0.017	**0.009**	**0.009**
6	1 – 10	0.045	0.006	0.019	0.019	**0.010**	**0.010**	0.201	**0.006**	0.059	0.017	**0.006**	**0.006**

Figura 4-3: Predicción con una red CC regularizada con la varianza para la serie de linces canadienses. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



4.3.3 Tercer caso: Manchas Solares

Esta serie contiene el número anual de las manchas solares sobre la cara del sol durante el período de 1700 a 1956, para un total de 256 datos. Esta serie ha sido estudiada por varios investigadores; entre ellos se encuentran (Zhang G. , 2003), (Cottrell, Girard, Girard, Mangeas, & Muller, 1995), y (De Groot & Wurtz, 1991). Esta serie se caracteriza por ser no lineal y ha sido usada tradicionalmente para medir la efectividad de modelos estadísticos no lineales (Ghiassi, Saidane, & Zimbra, 2005).

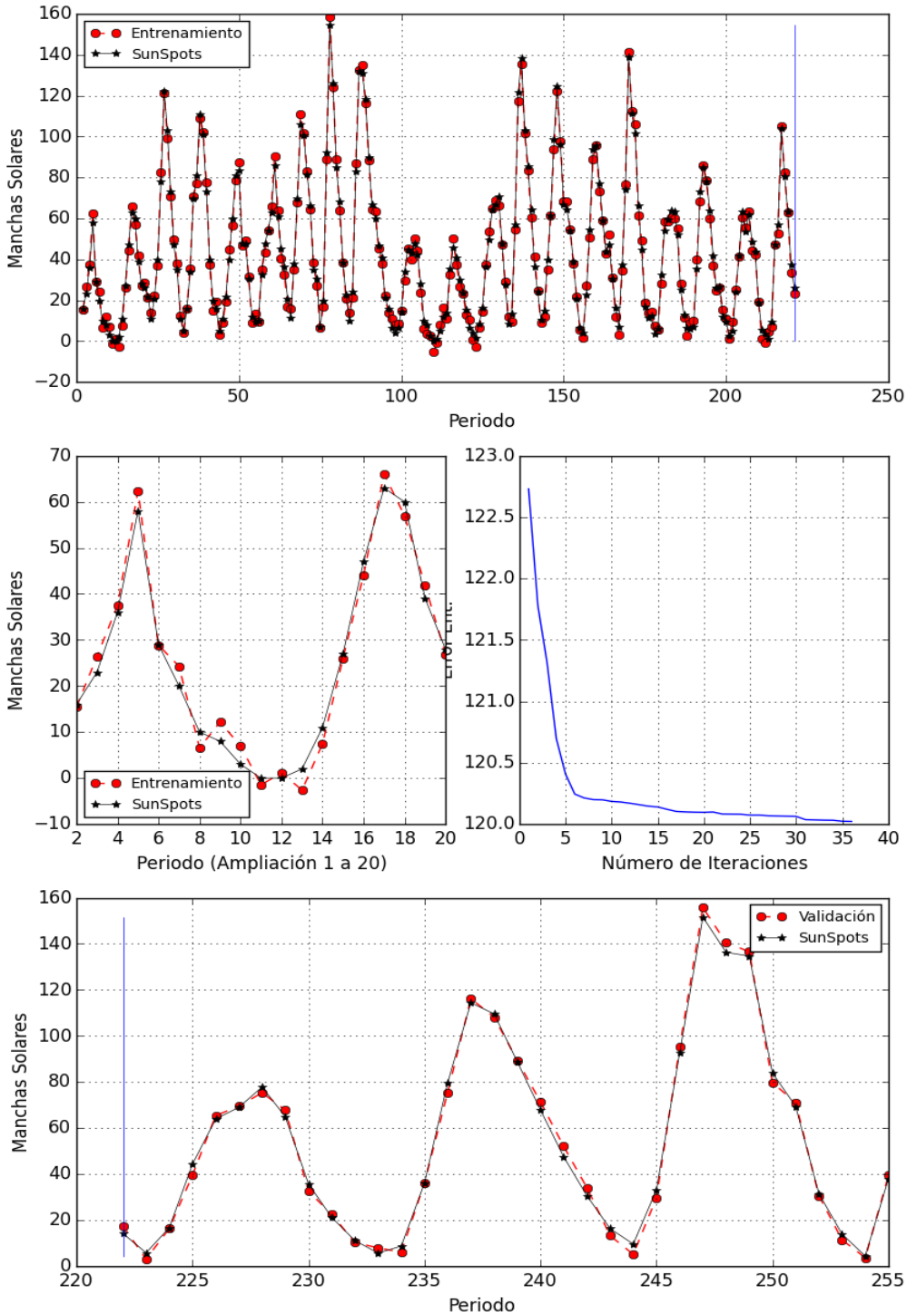
Como Ghiassi *et al.* (2005), se usa de las 256 observaciones de la serie, las primeras 221 para entrenamiento y las últimas 35 para validación; reportaron los resultados del DAN2 durante las fases de entrenamiento y validación al pronosticar con ésta serie con los MLP presentados en la Tabla 4-3.

Tabla 4-3: Valores del SSE para diferentes modelos pronosticando la serie del tercer caso.

Núm. Mod.	Rezagos	MSE Entrenamiento							MSE Validación						
		ANN	MLP	DAN2	CC	CC-DP	CC-VA	CC-DE	ANN	MLP	DAN2	CC	CC-DP	CC-VA	CC-DE
1	1,2,3,4	123	115	–	95	94	**93**	**93**	129	115	–	105	101	**98**	**102**
2	1,2,9,11	135	124	95	105	97	**93**	**94**	–	210	146	165	105	**76**	**78**
3	1 – 11	–	134	103	113	96	**93**	**94**	–	241	127	175	109	**75**	**85**
4	1,2,3,9,10,11	–	156	114	114	97	**90**	**92**	–	265	133	147	108	**79**	**78**
5	1,3,4,9,10,11	–	110	**78**	97	99	90	90	–	277	145	202	108	**79**	**78**

En la Tabla 4-3 se resumen los valores del MSE obtenido, tanto para entrenamiento como para validación, para los modelos estimados mediante CC. En la misma tabla puede observarse que con los modelos CC-VA y CC-VE se obtienen mejores resultados que con los MLP y en la mayoría de los casos que con DAN2. Además, para el modelo 2, el error obtenido con CC-VA es 11% y 77% menor que el logrado con DAN2 en entrenamiento y validación, respectivamente; mientras que los errores para el modelo 2 son 44.1% y 221.34% menores respecto al MLP. A partir de estos resultados se puede concluir que es conveniente utilizar CC-VA o CC-DE para pronosticar esta serie, ya que para la mayoría de los modelos se logró encontrar modelos con mejor capacidad de generalización que otros tradicionales.

Figura 4-4: Predicción con una red CC regularizada con la varianza para la serie de Manchas Solares. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



En la Figura 4-4 se presentan los valores reales de la serie de tiempo y los pronosticados con el modelo 3 de la Tabla 4-3. En ésta gráfica se puede observar que la serie posee crestas pronunciadas, que son difíciles de modelar; sin embargo, la red CC-VA es capaz de alcanzar un buen estadístico de ajuste en comparación con el MLP y DAN2.

4.3.4 Cuarto caso: Usuarios autenticados en un servidor

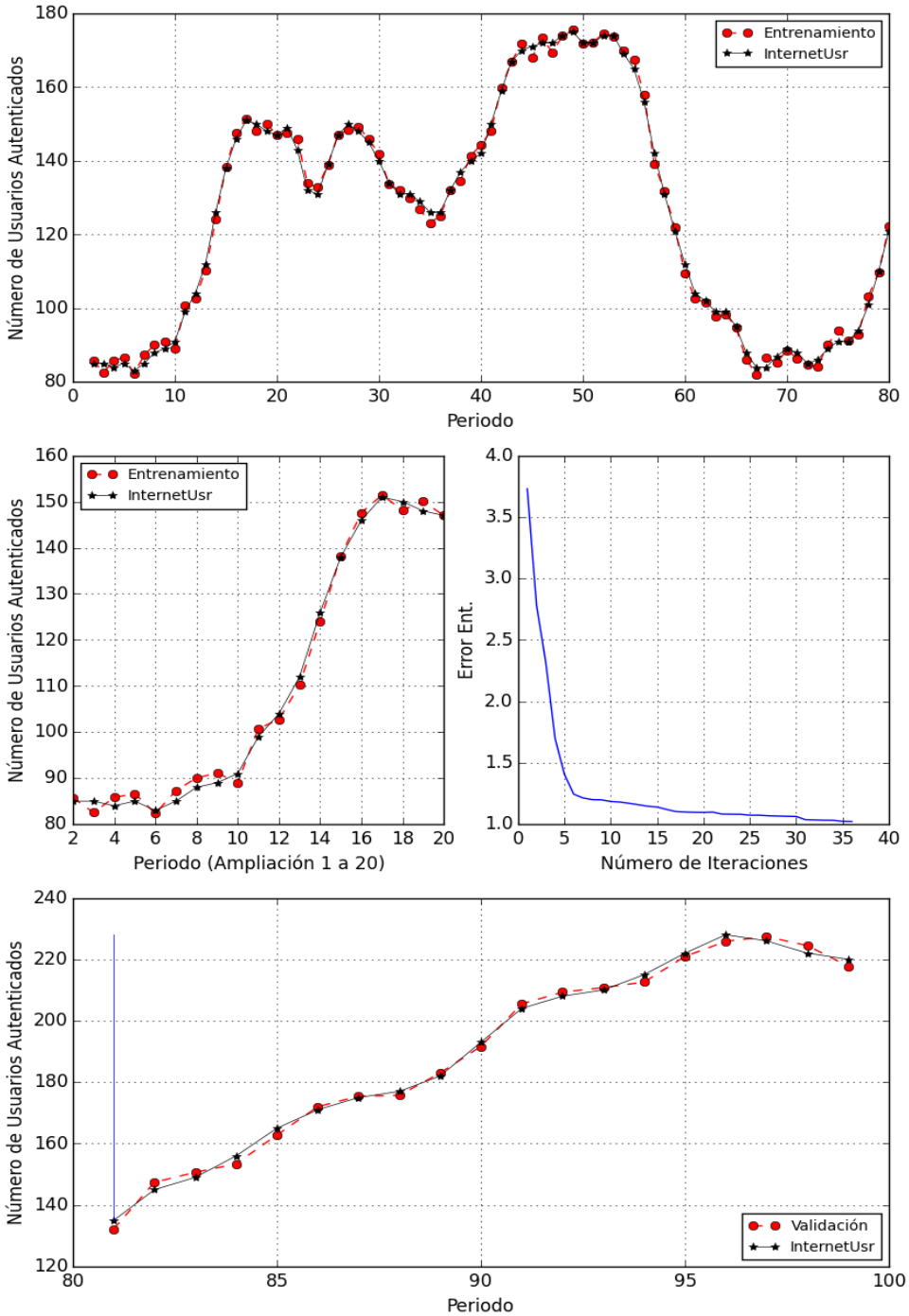
En ésta serie se registra el número de usuarios que iniciaron sesión en un servidor de Internet durante 100 minutos, para un total de 100 observaciones. Fue estudiada por Makridakis *et al.* (1998) mediante modelos ARIMA (Makridakis, Wheelwright, & Hyndman, 1998), Ghiassi et al. (2005) mediante *Dinamic Artificial Neural Network* (DAN2) y *Artificial Neural Network* (ANN) (Ghiassi, Saidane, & Zimbra, 2005); los resultados obtenidos por estos investigadores se resumen en la Tabla 4-4. Además, según el análisis realizado por Makridakis *et al.* (1998), esta serie es "no-estacionaria" en su proceso. Para los experimentos, de los 100 datos se tomaron los primeros 80 para entrenamiento, y el restante para validación, tal como se realizó en (Ghiassi, Saidane, & Zimbra, 2005).

Tabla 4-4: Valores del MSE al pronosticar la serie del cuarto caso con varios modelos.

Modelo	Rezagos	Entrenamiento	Validación
ARIMA	1, 2, 3, 4	9.760	8.110
ANN	1, 2, 3, 4	7.000	9.250
DAN2	1, 2, 3, 4	2.780	3.870
MLP	1, 2, 3, 4	5.764	5.634
CC	1, 2, 3, 4	7.299	6.971
CC-DP	1, 2, 3, 4	2.746	1.825
CC-VA	1, 2, 3, 4	2.650	1.198
CC-DE	**1, 2, 3, 4**	**2.512**	**1.025**

En la Tabla 4-4 se resumen los resultados, tanto en entrenamiento como en validación, al pronosticar la serie de este caso con una red MLP y con los tipos de modelos CC. Se puede observar que el menor error fue obtenido con el modelo CC-DE, es decir, cascada correlación regularizado con la desviación estándar. Los errores conseguidos con CC-DE son 9.5% y 73.51% menores que los logrados por DAN2, en entrenamiento y validación, respectivamente. Se nota una diferencia considerable en la validación, evidenciando que la red CC-DE tiene una capacidad de generalización superior a DAN2, incluso mejor que el resto de modelos. En la Figura 4-5 también se evidencia la excelente capacidad de generalización del modelo CC-DE.

Figura 4-5: Predicción con una red CC regularizada con la desviación estándar para la serie de Usuarios autenticados en un servidor. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



4.3.5 Quinto caso: Equipos de contaminación

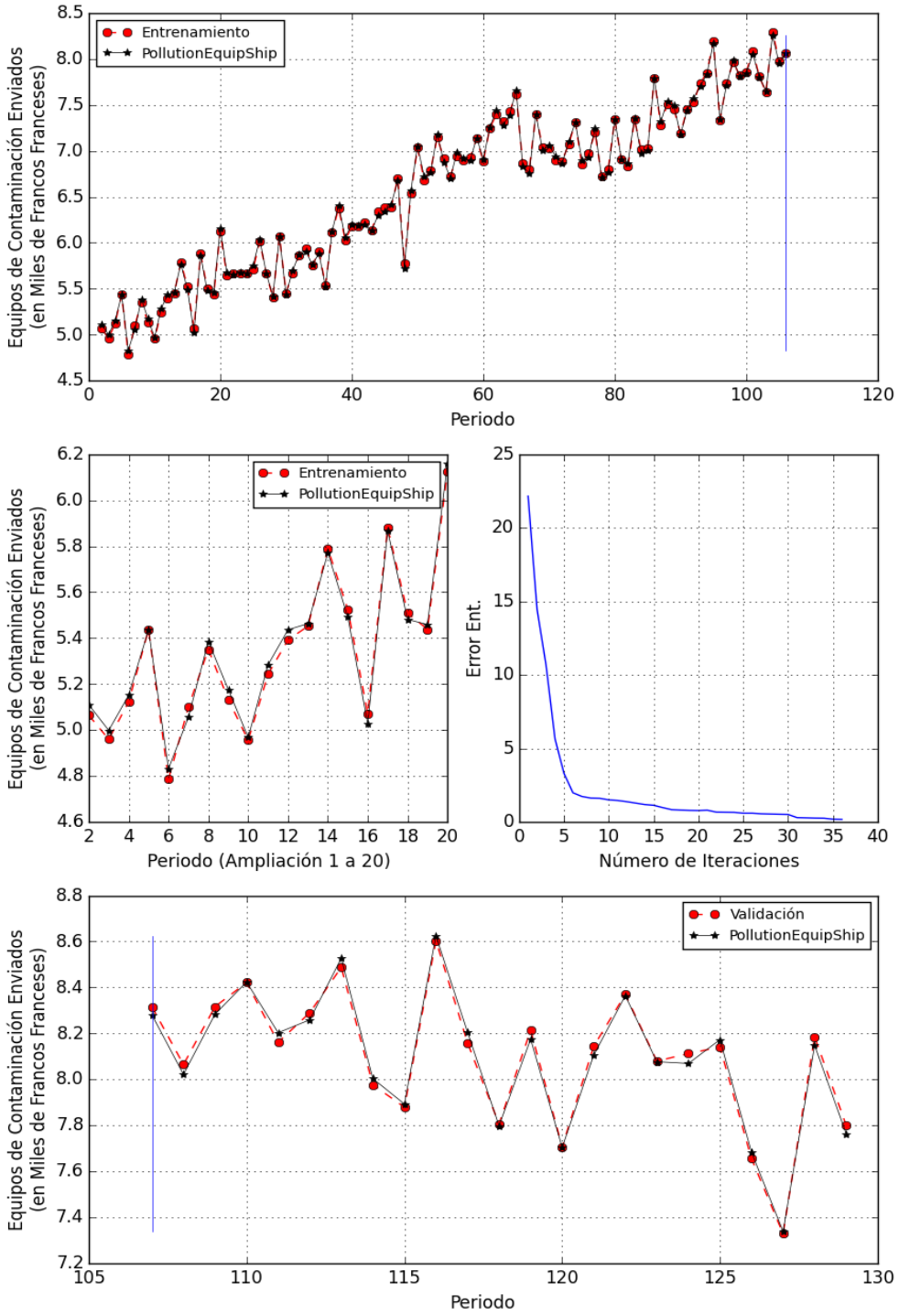
Esta serie contiene, en miles de Francos franceses, los envíos mensuales de equipos de contaminación desde Enero de 1986 hasta Octubre de 1996, para tener un total de 130 datos. Esta ha sido estudiada por diversos autores, entre estos, Makridakis *et al.* (1998) quien determinó que esta serie es "no-estacionaria en varianza" y usó ARIMA para pronosticarla; mientras que Ghiassi *et al.* (2005) la pronosticó mediante DAN2). En la Tabla 4-5 se resumen los resultados del MSE reportados en la literatura, los modelos correspondientes entre ARIMA y DAN2-3; además, se presentan los resultados obtenidos al pronosticar con MLP y las variantes de CC. Se usaron los primeros 106 datos como conjunto de entrenamiento, el resto para validación.

Tabla 4-5: Valores del MSE al pronosticar la serie del quinto caso con varios modelos.

Modelo	Rezagos	Entrenamiento	Validación
ARIMA	1,2,3,12,13,14,15	0.052	0.268
ANN	1 – 12	0.054	0.146
DAN2-1	1,2,3,12,13,14,15	0.020	0.025
DAN2-2	1,2,3,12,13,14,15	0.019	0.020
DAN2-3	1 – 15	0.013	0.023
MLP	1 – 15	0.045	0.120
CC	1 – 15	0.030	0.071
CC-DP	1 – 15	0.022	0.031
CC-VA	1 – 15	**0.012**	**0.019**
CC-DE	1 – 15	0.015	0.019

Los resultados continúan reflejando que tanto CC-VA como CC-DE tienen mejor capacidad de generalización que los demás modelos, de los dos el que tiene mejor capacidad de entrenamiento y de generalización es CC-VA, sin embargo, la diferencia con CC-DE es evidentemente mínima. La comparación del MSE entre CC-VA y DAN2-3 (el mejor de los reportados), es que tanto el error de entrenamiento y validación de CC-VA respecto DAN2-3 mejoró 7.69% y 17.39%, respectivamente. En la Figura 4-6, se presentan los valores reales y pronosticados de la serie, usando el modelo CC-VA, la gráfica evidencia la adecuada capacidad de generalización que ha logrado el modelo.

Figura 4-6: Predicción con una red CC regularizada con la varianza para la serie de equipos de contaminación. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



4.3.6 Sexto caso: Venta de Papel

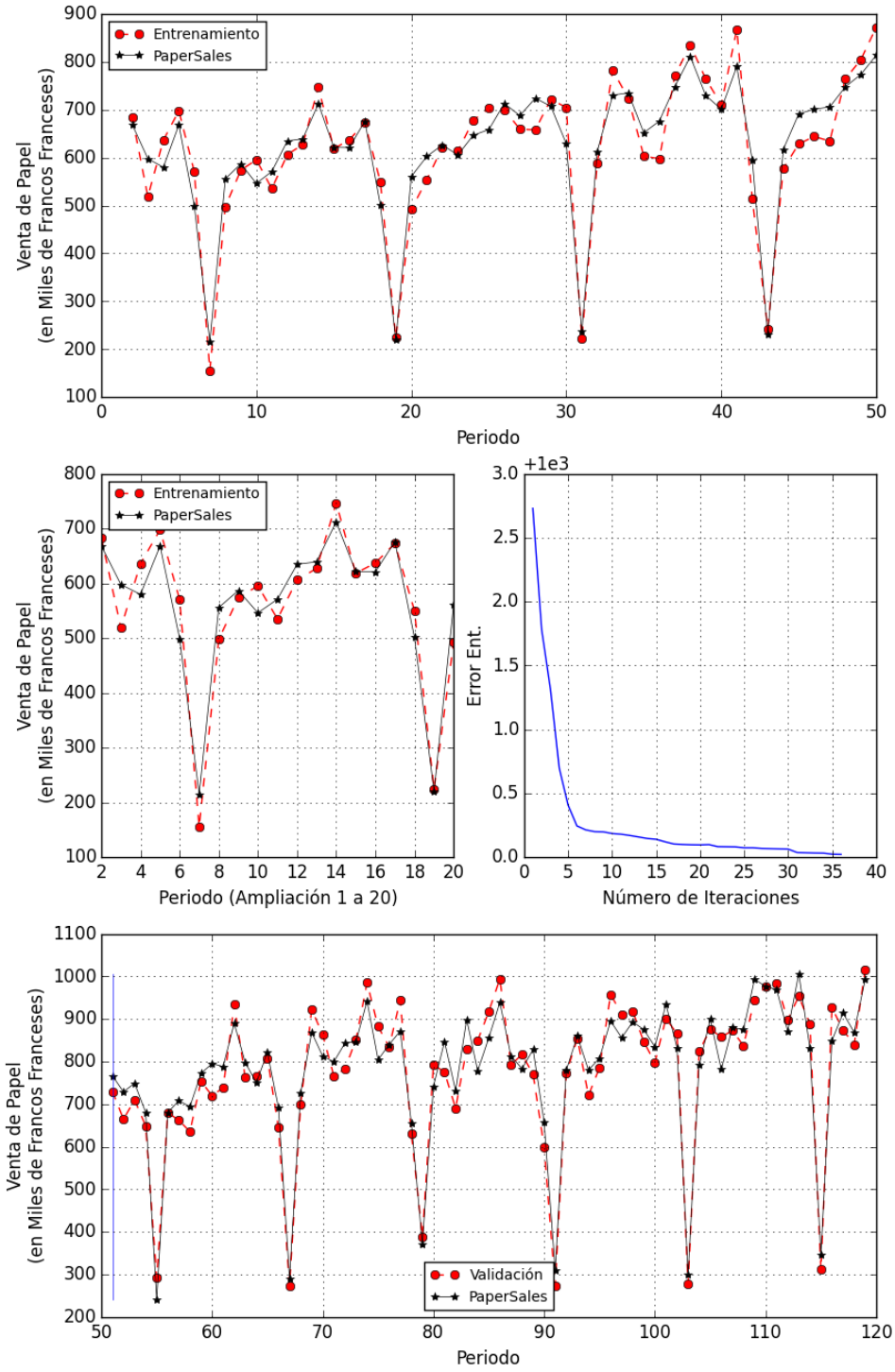
Esta serie contiene 120 observaciones, las cuales corresponden a las ventas de la industria, en miles de Francos franceses, de papel para imprimir y escribir entre enero de 1963 y diciembre de 1972. Para el entrenamiento se usaron las primeras 100 observaciones y las últimas 20 para validación. Makridakis *et al.* (1998) mostraron que esta serie es "estacional" con tendencia creciente y usó ARIMA para pronosticarla; también fue usada por Ghiassi *et al.* (2005) con DAN2. Los resultados obtenidos por los autores se reportan en la Tabla 4-6, desde el Modelo ARIMA hasta el DAN2-3.

Tabla 4-6: Valores del MSE al pronosticar la serie del sexto caso con varios modelos.

Modelo	Rezagos	Entrenamiento	Validación
ARIMA	1, 12, 13	1715.5	4791.7
ANN	1, 7, 12	1951.5	5874.8
DAN2-1	1, 2	1448.7	1782.7
DAN2-2	1, 12, 13	1671.4	2011.0
DAN2-3	1, 7, 12	1263.4	1760.8
MLP	1, 7, 12	1789.5	3548.2
CC	1, 7, 12	1546.2	2654.3
CC-DP	1, 7, 12	1354.2	1984.5
CC-VA	1, 7, 12	1248.2	1542.1
CC-DE	1, 7, 12	**1135.1**	**1495.2**

Los resultados experimentales se resumen en la Tabla 4-6, se continúa teniendo evidencia de que los modelos CC-VA y CC-DE son superiores a los demás, el mejor ha sido CC-DE, su diferencia en entrenamiento respecto a DAN2-3 es de 10.16% de mejora, mientras que en entrenamiento mejoró aún más en 15.1%. Esta serie es particularmente difícil de pronosticar, especialmente por sus "picos" los cuales evidentemente hacen que la tarea sea compleja; sin embargo, los modelos CC-VA y CC-DE son efectivos para pronosticar esta serie. La complejidad de la serie y el buen pronóstico de la misma se pueden evidenciar en la Figura 4-7.

Figura 4-7: Predicción con una red CC regularizada con desviación estándar para la serie de venta de papel. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



4.3.7 Séptimo caso: Mortalidad por diabetes.

El modelado y pronóstico de series de tiempo en salud es una herramienta que apoya a los agentes tomadores de decisiones del contexto de salud (*e.g.* Directores de Seccionales de Salud, Gerentes de EPS o IPS, Ministro de Salud) para la realización de la planificación de los servicios en salud, la regulación del mercado, vigilancia epidemiológica, establecimiento y evaluación de políticas, entre otros aspectos y eventos propios del sistema de salud de un país (Grisales R., Hoyos G., López J., Hincapié P., & Bello P., 2001).

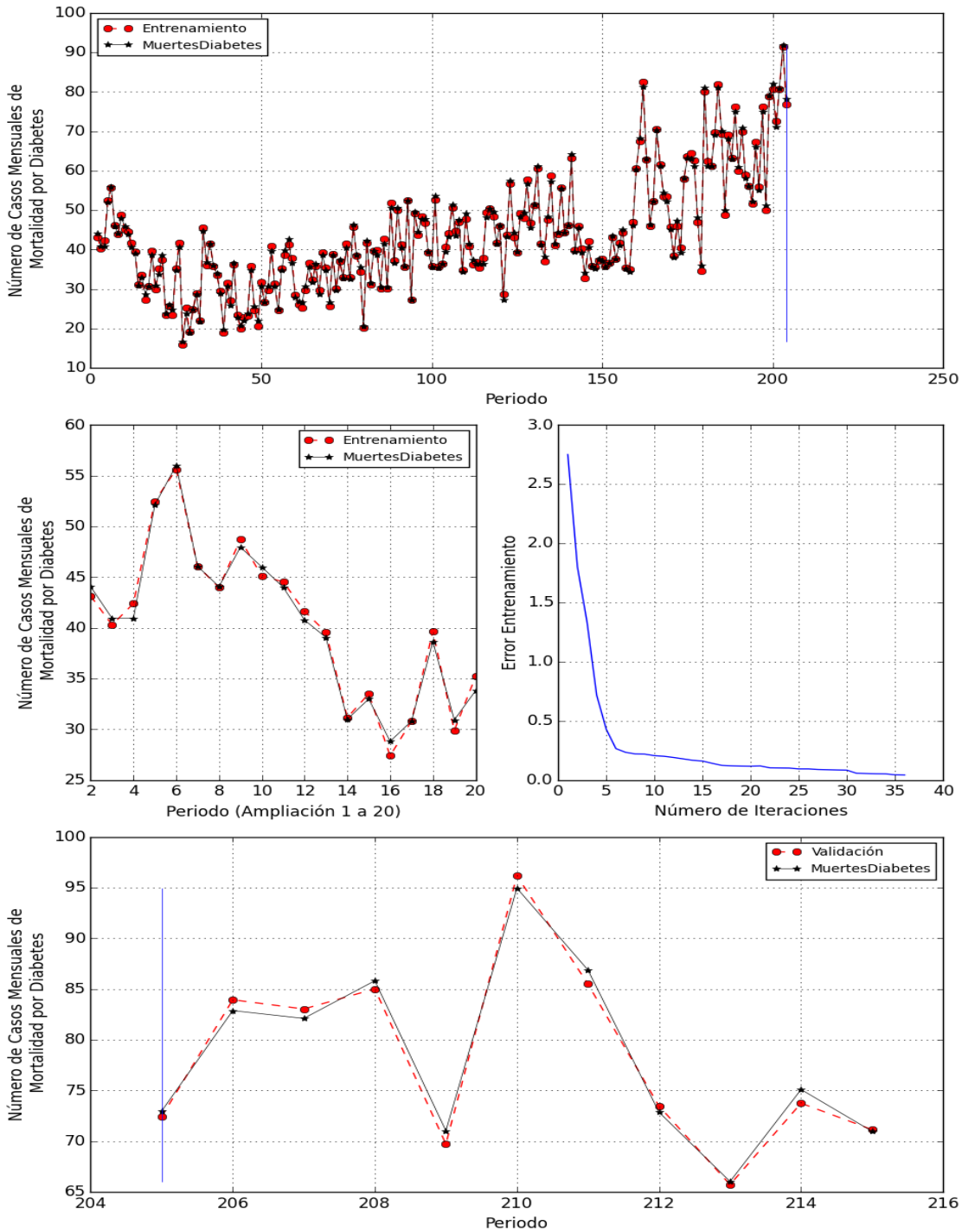
Específicamente, en salud es de especial interés la comprensión de los patrones de morbilidad y mortalidad, Grisales *et al.* (2001) analizaron diversas series de tiempo de eventos mensuales de mortalidad y morbilidad en Bogotá para los años entre 1982 y 1999 mediante modelos ARMA y ARIMA; para efectos de este trabajo, se toma la serie de tiempo del número mensual de casos de mortalidad por diabetes mellitus entre enero de 1982 y diciembre de 1999, consta de 216 observaciones, de las cuales se usan las primeras 204 (17 años) para entrenamiento y las últimas doce para validación (1 año).

En la Tabla 4-7 se presentan los resultados del MSE, tanto en entrenamiento como en validación, al pronosticar la serie de este caso con los modelos indicados; estos evidencian nuevamente excelentes resultados, para CC-VA y CC-DE respecto a MLP, CC y CC-DP, toda vez que tienen mejor capacidad de generalización. Además, estos resultados han demostrado, que el pronóstico de series de tiempo de salud pública con redes neuronales artificiales es posible y es una herramienta que debe ser tenida en cuenta por sus agentes. Finalmente, en la Figura 4-8 se continúa exhibiendo la excelente capacidad de generalización que se puede alcanzar con los modelos CC-VA.

Tabla 4-7: Valores del MSE al pronosticar la serie del sexto caso con varios modelos.

Modelo	Entrenamiento	Validación
MLP	0.032	0.051
CC	0.015	0.029
CC-DP	0.010	0.015
CC-VA	**0.005**	**0.009**
CC-DE	0.009	0.010

Figura 4-8: Predicción con una red CC regularizada con la varianza para la serie de mortalidad mensual por diabetes mellitus. Entrenamiento de la red (imagen superior), ampliación del entrenamiento y error de entrenamiento (imágenes centrales) y validación (imagen inferior)



4.4 Análisis estadístico de la precisión de pronóstico de los modelos

Aunque hasta este punto las redes CC-VA y CC-DE han permitido obtener modelos con una buena capacidad de generalización, incluso mejor que la de otros modelos reportados en la literatura, se requiere realizar un análisis de significancia estadística con el fin de presentar evidencia contundente sobre la efectividad de estos modelos. Para esto se usará el estadístico Morgan Granger Newbold (MGN) (Diebold & Mariano, 1995), con el cual se probará la hipótesis nula de que no hay diferencia entre la precisión de CC-VA y los demás modelos que compiten (CC, CC-DP, CC-DE). El procedimiento se realizó análogamente al presentado por Ghiassi *et al.* (2005) para evidenciar la efectividad de DAN2.

Los resultados de la prueba MGN se presentan en la Tabla 4-8, al comparar la efectividad de las redes Cascada-Correlación regularizadas con la varianza respecto a las redes Cascada Correlación sin regularizar (CC), regularizadas con descomposición de pesos (CC-DP) y regularizadas con la desviación estándar (CC-DE); los resultados muestran que la diferencia entre CC respecto a CC y CC-DP es significativa por tanto el modelo CC-VA es superior a CC y CC-DP; además, CC-VA respecto a CC-DE la diferencia no es significativa, por tanto no se puede afirmar que el uno es superior al otro.

Entonces, la prueba MGM, ha permitido demostrar que los métodos de regularización propuestos son superiores a la red sin regularizar o regularizada con descomposición de pesos, esto le da confiabilidad a los modelos propuestos y evidencia su efectividad para realizar el pronóstico de series de tiempo.

Tabla 4-8: Resultados de la prueba MGN.

Modelo	***t-Stat* Calculado**	***t* Crítico**	**¿La Diferencia es Significativa?**
Entrenamiento Caso 4	*Usuarios autenticados en un servidor*		
CC *vs.* CC-VA	2.784	2.15	SI
CC-DP *vs.* CC-VA	2.986	2.15	SI
CC-DE *vs*. CC-VA	1.982	2.15	NO
Validación Caso 4			
CC *vs.* CC-VA	6.321	2.78	SI
CC-DP *vs.* CC-VA	6.654	2.78	SI
CC-DE *vs*. CC-VA	1.261	2.78	NO

Entrenamiento Caso 5	*Equipos de Contaminación*		
CC *vs.* CC-VA	1.89	1.23	SI
CC-DP *vs.* CC-VA	1.65	1.23	SI
CC-DE *vs*. CC-VA	1.01	1.23	NO
Validación Caso 5			
CC *vs.* CC-VA	1.98	1.32	SI
CC-DP *vs.* CC-VA	1.78	1.32	SI
CC-DE *vs*. CC-VA	0.98	1.32	NO
Entrenamiento Caso 6	*Venta de Papel*		
CC *vs.* CC-VA	2.18	1.56	SI
CC-DP *vs.* CC-VA	2.65	1.56	SI
CC-DE *vs*. CC-VA	1.25	1.56	NO
Validación Caso6			
CC *vs.* CC-VA	2.89	1.85	SI
CC-DP *vs.* CC-VA	2.15	1.85	SI
CC-DE *vs*. CC-VA	1.16	1.85	NO

4.5 Conclusiones

Los resultados experimentales permitieron evidenciar que la propuesta de CC-VA y CC-DE permite encontrar modelos con mejor capacidad de generalización; mientras que en el aprendizaje sin regularizar entre más se agregan parámetros más decae el error de entrenamiento, en el aprendizaje con regularización por más neuronas que se agreguen se conserva la superficie estable, conservando el mejor modelo encontrado. La estrategia de regularización, por su parte, permite encontrar un modelo tras el cual los valores del error se estabilizan, por lo que puede considerarse como el mejor modelo (*i.e.* adiciones posteriores no reflejan una ganancia en el error de ajuste).

Si bien, la regulación ha sido concebida como una técnica para resolver el problema de sobre-entrenamiento que a menudo adolecen los modelos de redes neuronales, esta también puede ser vista como estrategia para la selección del modelo, penalizando el número de parámetros del mismos.

Experimentalmente, se ha demostrado que la regularización mediante la varianza o la desviación estándar de los parámetros es una alternativa adecuada para la selección del modelo, toda vez que, conduce a resultados satisfactorios al permitir hallar un modelo con buen ajuste, buena capacidad de generalización y parsimonioso.

5. Conclusiones y recomendaciones

5.1 Cumplimiento de los objetivos propuestos

5.1.1 Objetivo Específico 1.

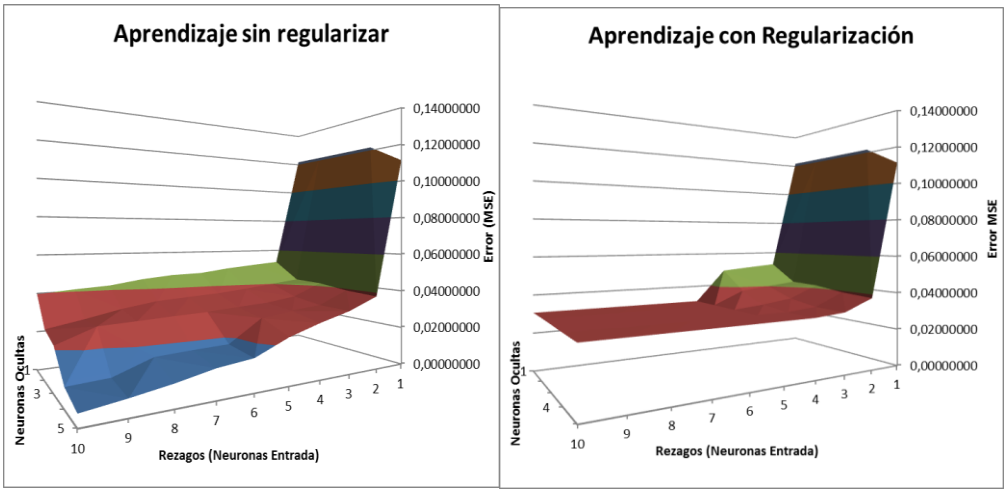
"*Determinar un criterio de información para las redes cascada correlación para la selección de modelos entre un conjunto finito"*

Los criterios de información son herramientas de selección de modelos que se pueden usar para comparar varios modelos de la misma arquitectura que se ajustan a los mismos datos. Básicamente, los criterios de información son medidas basadas en el ajuste del modelo que incluyen una penalización por complejidad, de este modo se requiere un criterio que considere la arquitectura especial de las redes cascada-correlación y permita seleccionar cuál es el modelo adecuado para representar una serie de tiempo.

Si bien la regularización dada por la Ecuación (1.9) ha sido concebida como un método para resolver problemas mal condicionados que permite controlar el sobreajuste, ésta también puede ser usada como estrategia para la selección del modelo, penalizando la función objetivo de entrenamiento y por ende penalizando los parámetros del mismo, lo cual permitirá seleccionar un modelo con adecuada capacidad de generalización. El método de regularización posee dos componentes críticos, el primero es la selección del parámetro de regularización y el segundo el término de regularización (i.e. Término de Penalización Compleja). El término de regularización indica cómo se debe penalizar la función objetivo, mientras que el parámetro indicará qué tanta incidencia tendrá ese término sobre el aprendizaje, es decir, qué tanto se debe penalizar para tener una adecuada capacidad de generalización.

Entonces, el parámetro de regularización (durante toda la tesis se trata como λ) es un indicador de suficiencia sobre el conjunto de entrenamiento usado, dado que si el parámetro es igual a cero, el entrenamiento se realizará sin restricciones, es decir, sin regularización; mientras que cuando el parámetro de regularización es diferente de cero y tiende a infinito, el término de regularización compleja dominará el aprendizaje, básicamente descartando los datos de entrenamiento. En la Figura 5-1, se evidencia que la estrategia de regularización permite encontrar modelos con mejor capacidad de generalización al controlar el sobreajuste en modelos sobreparametrizados; mientras que en el aprendizaje sin regularizar entre más se agregan parámetros más decae el error de entrenamiento y más aumentará el de validación. Además, en el aprendizaje con regularización por más neuronas que se agreguen se conserva la superficie estable, conservando el mejor modelo encontrado.

Figura 5-1. Contraste entre el aprendizaje sin regularizar y la regularización como criterio de información.



La estrategia de regularización, permite encontrar un modelo tras el cual los valores del error se estabiliza, por lo que puede considerarse como el mejor modelo, es decir, adiciones posteriores no reflejan una ganancia en el error de ajuste. Con el fin de comprender lo expuesto, en el Capítulo 1, se abordaron todas las consideraciones del aprendizaje predictivo, partiendo de las nociones, luego describiendo el sobreajuste como principal problema del aprendizaje en las redes neuronales y describiendo las técnicas más extendidas para controlarlo, las cuales a su vez pueden ser usadas como criterios de información.

Existen dos aportes fundamentales en la consecución de este objetivo: primero, el tipo de análisis planteado no ha sido realizado desde el punto de vista del marco de trabajo de aprendizaje predictivo para redes cascada-correlación, como tampoco se había tratado como un problema de optimización no-lineal mal condicionado para justificar el uso de la regularización en este tipo de redes cascada correlación; segundo, justificar conceptualmente, por qué usar la regularización como técnica para controlar el sobreajuste, esto permite que tal técnica se pueda plantear también como un criterio de información; y tercero, se plantearon las limitaciones de la regularización, que deben ser solucionadas para que el criterio sea efectivo, las cuales están relacionadas con la búsqueda de un valor de λ y la definición de un término de regularización o penalización compleja y se propone una solución en esta Tesis.

Con el fin de crear un criterio lo suficientemente robusto en el siguiente objetivo se incorpora una estrategia sistemática de selección del parámetro de regularización, que permitirá crear un nuevo criterio de selección de modelos.

5.1.2 Objetivo Específico 2.

"Determinar una estrategia sistemática de selección del parámetro de regularización para la red neuronal cascada correlación que permita controlar la complejidad del modelo."

En el objetivo anterior se expuso la regularización como una manera para controlar el sobreajuste que además es útil como criterio de información; sin embargo, existen limitaciones que se deben resolver, la primera es cómo seleccionar el parámetro λ y la segunda determinar un término de penalización compleja que controle efectivamente las magnitudes de los parámetros del modelo. Respecto a estas limitaciones se discuten ampliamente desde un punto de vista conceptual en la Sección 1.5, mientras que en el Capítulo 2, se muestra desde la perspectiva teórico conceptual y experimental tales limitaciones. Para solventarlas en el Capítulo 3 se propuso una nueva estrategia sistemática de selección de modelos en redes CC, que consiste en un proceso ordenado y estándar que se ajusta al aprendizaje constructivo de la arquitectura de la red CC que incorpora la regularización para obtener modelos con adecuada capacidad de generalización.

En el criterio, si dado un modelo su error de validación sin regularizar es menor que el mismo modelo regularizado (siendo λ relativamente pequeño) entonces no se requiere regularizar, y se puede continuar incrementando la complejidad del modelo. Si por el contrario el error de validación sin regularizar es mayor que el mismo modelo regularizado (siendo λ relativamente pequeño), desde ese modelo se requiere regularizar, por consiguiente al agregar más complejidad se deberá penalizar más (incrementar el valor de λ). Las consideraciones bajo las cuales fue diseñado el criterio están definidas en la Sección 3.2.1. Donde lo importante no es cuánto es el valor del parámetro de regularización, lo valioso es conocer desde cuándo se requiere regularizar, es decir, desde qué cantidad de neuronas se requiere que λ sea diferente de cero.

Entonces el principal logro de este objetivo es la definición de un nuevo criterio de selección de modelos, el cual fue evaluado experimentalmente y mostró tener la capacidad adecuada para seleccionar modelos con adecuada capacidad de generalización.

5.1.3 Objetivo Específico 3.

"Determinar una estrategia de regularización para la red neuronal cascada correlación que permita controlar efectivamente la magnitud de los parámetros del modelo."

Aún queda por resolver una limitación, cómo controlar efectivamente la magnitud de los parámetros del modelo, como se mostró en el Capítulo 2, la estrategia de regularización de descomposición de pesos no controla efectivamente la magnitud de los parámetros, lo cual es indispensable para poder controlar el sobreajuste. De este modo, en el Capítulo 3 se realiza el análisis teórico conceptual de dos cosas que es necesario penalizar y que deben estar explícitas en el criterio de regularización: primero la complejidad del modelo, dado por la cantidad de parámetros del mismo; y segundo la fluctuación o dispersión de los parámetros, lo cual se puede medir a través de la varianza o la desviación estándar.

Entonces un logro importante en este objetivo es definir la varianza y la desviación estándar como dos nuevas estrategias de regularización que controlan efectivamente la magnitud de los parámetros y que también penaliza la complejidad del modelo, esto se

comprobó experimentalmente la Sección 4.2. donde se llevó a cabo un experimento tradicional de la literatura mostrando que estas estrategias permiten encontrar modelos con adecuada capacidad de generalización y deben ser tenidas en cuenta en el conjunto de herramientas para controlar el sobreajuste.

5.1.4 Objetivo Específico 4

"Diseñar un criterio integral con base en los elementos definidos en los objetivos específicos 1, 2 y 3.

Ahora bien, la unión de los logros de los objetivos específicos uno, dos y tres, crea un nuevo criterio de selección de modelos lo suficientemente robusto que también permite controlar efectivamente el sobreajuste en las redes neuronales cascada-correlación. Su efectividad se evalúa en el capítulo 4.3, obteniendo excelentes resultados incluso superiores a los obtenidos por Ghiassi *et al.* (2005) con una arquitectura robusta como DAN2.

5.1.5 Objetivo General

"Diseñar un criterio para la selección del modelo de red neuronal cascada correlación que permita controlar integralmente el tamaño del conjunto de entrenamiento, la complejidad del modelo y la magnitud de pesos para obtener modelos con buena capacidad de generalización."

La consolidación y unión de los cuatro objetivos específicos conlleva a alcanzar el objetivo general. Aparte de lo ya mencionado, el criterio definido es novedoso e innovador dado que para poder llegar a tal solución se requirió hacer una construcción teórico-conceptual para poder generar nuevo conocimiento a partir del existente, lo cual garantizó en gran medida el éxito de este trabajo que además fue evidenciada en la práctica experimentalmente.

5.2 Contribuciones logradas

Los trabajos y contribuciones de esta tesis han sido difundidos mediante la presentación de ponencias en congresos y la publicación de artículos en revistas especializadas, los cuales se detallan a continuación:

Casos de aplicación, publicados en revistas indexadas:

1. "Predicción del precio de la electricidad en Brasil usando redes cascada-correlación". Villa, Fernán; Velásquez, Juan; Revista U.D.C.A Actividad & Divulgación Científica. 14(2). p.p. 161-167. 2011.
2. "Pronóstico de series de tiempo con redes neuronales regularizadas y validación cruzada". Velásquez, Juan; Fonnegra, Yulieth; Villa, Fernán; Revista Vínculos. 10(1). P.p. 267 – 279. 2013.
3. "Control de sobreajuste en redes neuronales tipo cascada-correlación aplicado a la predicción de precios de contratos de electricidad". Villa, Fernán; Sánchez, Paola; Velásquez, Juan. Revista Ingenierías Universidad de Medellín. 14(26). 2014.

Congresos Nacionales e Internacionales:

1. MAEB'12 – VIII Congreso Español sobre Metaheuristicas, Algoritmos Evolutivos y Bioinspirados. "Estimación de parámetros para redes cascada correlación utilizando Supernova". Eddy Mesa, Fernán Villa, Juan David Velásquez, Gloria Patricia Jaramillo. España. Febrero, 2012.
2. ISF 2012 – International Symposium of Forecasting. "Regularized Cascade Correlation Networks for Electricity Spot Price Forecasting in Colombia", Fernán Villa, Juan David Velásquez. Boston, MA. June, 2012. **Fue ganador de ISF 2012 Travel Award por la calidad del trabajo presentado.**
3. CICOM 2013 - 3° Congreso Internacional de Computación México - Colombia Y XIV Jornada Académica En Inteligencia Artificial. "Pronóstico de series de tiempo con redes neuronales regularizadas y validación cruzada". Fernán Villa, Juan David Velasquez, Yulieth Fonnegra. Septiembre, 2013.

4. ICAMI 2013 - International Conference on Applied Mathematics and Informatics. “Regularized cascade correlation networks for time series forecasting”. Fernán Villa, Juan David Velasquez, Yulieth Fonnegra. Noviembre, 2013.

Respecto a la implementación de los modelos planteados en esta tesis, se desarrollaron en el lenguaje Python usando como entorno integrado de desarrollo Eclipse, lo cual facilita el ciclo de vida del software, especialmente las fases de pruebas, validación y mantenimiento. Este lenguaje ha tenido gran acogida tanto en el ambiente académico como en la industria, evidencia de esto, es la séptima posición en el índice TIOBE (http://www.tiobe.com/) ocupada en el último año, por encima de lenguajes como R o Matlab ®. Desde el punto de vista académico, algunas de las ventajas de Python pueden consultarse en (Shen, 2014). El paquete de librerías desarrollado recibió el nombre de *netfor* (*neural networks for forecasting*) al momento de escribir esta tesis se encuentra en la versión 2.4.7.

Otra contribución de esta tesis ha sido el diseño de una línea de investigación en “*Inteligencia Computacional en Salud*” en la Facultad Nacional de Salud Pública, Universidad de Antioquia, siendo de principal interés temas relacionados explícitamente son el análisis, modelado y pronóstico de series de tiempo epidemiológicas. Además, tiene como objetivo proponer y establecer soluciones desde las técnicas y métodos computacionales a problemas relacionados con la salud pública (*e.g.* medición de desigualdades, pronósticos de contaminación, detección temprana y prevención de enfermedades, evaluación de políticas).

Además, se ha incorporado al currículo de la carrera de “Gerencia de Sistemas de Información de Salud” tres cursos electivos que conforman la línea de “Inteligencia Computacional en Salud”, estos cursos son: “Pronóstico de Series de Tiempo”, “Clasificación de Patrones” y “Simulación de Sistemas”. El curso de Pronóstico de Series de Tiempo, es producto de esta tesis e incorpora los avances teórico-prácticos desarrollados en esta Tesis.

Bibliografía

Anders, U., & Korn, O. (1999). Model selection in neural networks. *Neural Networks*(12), 309-323.

Argüelles Pabón, D. C., & Nagles García, N. (2009). *Estrategias para promover procesos de aprendizaje autónomo.* Bogotá: Universidad EAN.

Awad, E. M., & Ghaziri, H. M. (2008). *Knowledge Management* (Segunda ed.). India: Pearson Education.

Bishop, C. M. (1994). *Neural Networks for Pattern Recognition.* New York: Oxford University Press Inc.

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition.* Oxford: Oxford University Press.

Bollerslev, T. (1986). Generalised autoregressive conditional heteroscedasticity. *Journal of Econometrics, 31*, 307-327.

Bowerman, B. L., O'Connell, R. T., & Koehler, A. B. (2006). *Forecasting, Time Series, and Regression: An Applied Approach* (Cuarta ed.). Ohio: Cengage Learning Brooks Cole.

Box, G., & Jenkins, G. (1976). *Time series analysis, forecasting and control.* San Francisco, Holden-Day.

Burger, M., & Neubauer, A. (2003). Analysis of Tikhonov regularization for function approximation by neural networks. *Neural Networks, 16*(1), 79-90 .

Campbell, M. J., & Walker, A. M. (1977). A survey of statistical work on the mackenzie river series of annual canadian lynx trappings for the years 1821–1934 and a new analysis. *Journal of the Royal Statistical Society, 140*(4), 411–431.

Cao, L., & Tay, F. (2003). Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Transactions on Neural Networks, 14*(6), 1506-1518.

Chan, K. S., & Tong, H. (1986). On estimating thresholds in autoregressive models. *Journal of Time Series Analysis*(7), 178-190.

Cherkassky, V. (2012). Predictive Learning, Knowledge Discovery and Philosophy of Science. En J. Liu, C. Alippi , B. Bouchon-Meunier, G. Greenwood, & H. Abbass, *Advances in Computational Intelligence. Lecture Notes in Computer Science* (Vol. 7311, págs. 209-233). Berlin Heidelberg: Springer.

Cherkassky, V., & Ma, Y. (2006). Data Complexity, Margin Based Learning. En M. Basu, & T. Kam Ho (Edits.), *Data Complexity in Pattern Recognition. Advanced Information and Knowledge Processing* (págs. 91-114). London: Springer.

Cherkassky, V., & Mulier, F. M. (2007). *Learning from Data: Concepts, Theory, and Methods.* New Jersey: WILEY.

Chow, T. W., & Cho, S. (2007). *Neural networks and computing: learning algorithms and applications* (Vol. 7). London: Imperial College Press.

Chow, T., & Cho, S.-Y. (2007). *Neural networks and computing: learning algorithms and applications* (Vol. 7). London: Imperial College Press.

Cios, K., & Pedrycz, W. (1998). Data Mining Methods for Knowledge Discovery. *IEEE Transactions on Neural Networks, 9*(6), 1533 - 1534.

Cogollo, M., & Velasquez, J. (2014). Methodological Advances in Artificial Neural Networks for Time Series Forecasting. *Latin America Transactions, IEEE (Revista IEEE America Latina), 4*(2), 764 - 771.

Cottrell, M., Girard, B., Girard, Y., Mangeas, M., & Muller, C. (1995). Neural modeling for time series: A statistical stepwise method for weight elimination. *IEEE Transactions on Neural Networks, 6*(6), 1355–1364.

Crone, S., & Kourentzes, N. (2009). Input-variable Specification for Neural Networks - An Analysis of Forecasting low and high Time Series Frequency. (I. N. York, Ed.) *Proceedings of the International Joint Conference on Neural Networks, IJCNN'09*, pág. in press.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control: Signals and Systems, 2*, 202–314.

De Gooijer, I., & Kumar, K. (1992). Some Recent Developments in Non- Linear Modelling, Testing, and Forecasting. *International Journal of Forecasting, 8*, 135-156.

De Groot, C., & Wurtz, D. (1991). Analysis of univariate time series with connectionist nets: A case study of two classical examples.Neurocomputing. *Neurocomputing, 3*, 177–192.

Dick van Dijk, D., Teräsvirta, T., & Franses, F. (2002). Smooth Transition Autoregressive Models - A Survey Of Recent Developments. *Econometric Reviews*(21), 1-47.

Diebold, F. X., & Mariano, R. S. (1995). Comparing Predictive Accuracy. *Journal of Bussiness and Economics Statistics, 13*(3), 253-263.

Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern Classification* (Second ed.). New York, United States of America: Wiley-Interscience Publication.

Dutoit, X., Schrauwen, B., Van Campenhout, J., Stroobant, D., Van Brussel, H., & Nuttin, M. (2009). Pruning and Regularization in Reservoir Computing. *Neurocomputing*(72), 1534 - 1546.

Engle, R. (1982). Autoregressive conditional heteroskedasticity with estimates of the variance of UK inflation. *Econometrica, 50*, 987–1008.

Evgeniou, T., Poggio, T., Pontil, M., & Verri, A. (2002). Regularization and Statistical Learning Theory for Data Analysis. *Computational Statistics & Data Analysis*(38), 421-432.

Evgeniou, T., Pontil, M., & Poggio, T. (1999). *A unified framework for Regularization Networks and Vector Support Machines.* Massachusetts Institute of Technology: Artificial Inteligence Laboratory.

Fahlman, S. E., & Lebiere, C. (1990). The Cascade-Correlation Learning Architecture. *Advances in Neural Information Processing Systems , 2*, 524-532.

Faraway, J., & Chatfield, C. (1998). Time series forecasting with neural networks: A comparative study using the airline data. *Applied Statistics, 47*(2), 231–250.

Flórez López, R., & Fernández, J. M. (2008). *Las Redes Neuronales Artificiales, Fundamentos Teóricos y Aplicaciones Prácticas.* España: Netbiblo.

Friedman, J. (1994). An overview of predictive learning and function approximation. En V. Cherkassky, J. Friedman, & H. Wechsler (Edits.), *Statistics to Neural Networks* (NATO ASI Series F ed., pág. 136). New York: Springer.

Funahashi, K. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Neworks, 2*, 183–192.

Garrett, J. J. (2001). *Introducción a la Arquitectura de la Información*. Recuperado el 5 de 4 de 2015, de http://mantruc.com/palabras/intro-ia/mod-jjg.html

Gençay, R., & Liu, T. (1997). Nonlinear modelling and prediction with feedforward and recurrent networks. *Physica D: Nonlinear Phenomena, 108*(1-2), 119-134 .

Ghiassi, M., Saidane, H., & Zimbra, D. (2005). A dynamic artificial neural network model for forecasting time series events. *International Journal of Forecasting, 21*(2), 341-362.

Girosi, F., Jones, M., & Poggio, T. (1995). Regularization Theory and Neural Networks Architectures. *Neural Computation, 7*(2), 219-269.

Giustolisi, O., & Laucelli, D. (2005). Improving generalization of artificial neural networks in rainfall-runoff modelling. *Hydrological Sciences–Journal–des Sciences Hydrologiques, 3*(50), 439 - 457.

Granger, C. (1993). Strategies for modelling nonlinear time-series relationships. *The Economic Record, 69*(206), 233–238.

Granger, C., & Anderson, A. (1978). *An Introduction to Bilinear Time Series Models.* Gottingen: Vandenhoeck and Ruprecht.

Granger, C., & Teräsvirta, T. (1993). *Modelling Nonlinear Economic Relationships.* Oxford: Oxford University Press.

Grisales R., H., Hoyos G., C., López J., A. M., Hincapié P., D., & Bello P., L. D. (2001). *Análisis de series de tiempo de eventos de mortalidad y morbilidad, Bogotá 1982 - 1999.* Bogotá, Colombia: Secretaría Distrital de Salud.

Hagiwara, K. (2002). Regularization Learning, Early Stopping and Biased Estimator. *Neurocomputing*(48), 937-955.

Hansen, C. (1998). *Rank-Deficient and Discrete Ill-Posed Problems - Numerical Aspects of Linear Inversion.* Pensilvania, USA: SIAM.

Hardle, W., Liitkepohl, H., & Chen, R. (1997). A review of non-parametric time series analysis. *Int. Statist. Rev., 65*, 49-72.

Hawkins, D. M. (2004). The Problem of Overfitting. *J. Chem. Inf. Comput. Sci.*(44), 1-12.

Haykin, S. (1994). *Neural Networks, A Comprehensive Foundation.* New York: Macmillan College Publishing Company, Inc.

Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation.* New Jersey: Prentice Hall.

Hinton, G. (1989). Connectionist learning procedures. *Artificial Intelligence*(40), 185–243.

Hoerl, A. E., & Kennard, R. W. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics., 12*(1), 55–67.

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

Hush, D., & Horne, B. (1993). Progress in supervised neural networks. *Signal Processing Magazine, IEEE, 10*(1), 8-39. Recuperado el 19 de 08 de 2013, de http://www.oni.escuelas.edu.ar/olimpi97/literatura-argentina/Autores/S%C3%A1bato/Sabato.htm

Hyndman, R., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *Journal of Statistical Software, 26*(3).

Ishikawa, M. (1989). *A structural learning algorithm with forgetting of link weights.* Japan: Electrotechnical Laboratory.

Ishikawa, M. (1996). Structural Learning with Forgetting. *Neuralnetworks, 9*(3), 509-521.

Jin Cui, Y., Davis , S., Cheng, C.-K., & Xue, B. (2004). A study of sample size with neural network. *Proceedings of 2004 International Conference on Machine Learning and Cybernetics, 6*, 3444 - 3448.

Kaastra, I., & Boyd, M. (1996). Designing a neural network for forecasting financial and economic series. *Neurocomputing*(10), 215-236.

Kadogiannis, V., & Lolis, A. (2002). Forecasting financial time series using neural network and fuzzy system-based techniques. *Neural Computing and Application, 11*, 90-102.

Karystinos, G., & Pados, D. (2000). On overfitting, generalization, and randomly expanded training sets. *IEEE Transactions on Neural Networks, 11*(5), 1050-1057 .

Kasabov, N. (1998). *Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering* (2da ed.). Massachusetts: The MIT Press Cambridge.

Kitchenham, B. (2004). *Procedures for Undertaking Systematic Reviews.* P.Joint Technical Report, Computer Science Department, Keele University (TR/SE- 0401) and National ICT Australia Ltd. (0400011T.1).

Lachtermacher, G., & Fuller, J. (1995). Backpropagation in time-series forecasting. *Journal of Forecasting, 14*, 381–393.

Lang, K., Waibel, A., & Hinton, G. (1990). A time-delay neural network architecture for isolated word recognition. *Neural Networks*(3), 23 - 43.

Larose, D. T., & Larose, C. D. (2015). *Data Mining and Predictive Analytics (Wiley Series on Methods and Applications in Data Mining)* (Segunda ed.). NY: Wiley.

Le Cun, Y., Denker, J. S., & Solla, S. A. (1990). Optimal brain damage. En D. S. Touretzky (Ed.), *Advances in Neural Information Processing Systems* (Vol. 2, págs. 598–605). San Mateo, California, USA: Morgan Kaufmann.

Leung, C., Wang, H., & Sum, J. (2010). On the selection of weight decay parameter for faulty networks. *IEEE Transactions on Neural Networks , 8*(21), 1232-1244 .

Leung, C.-T., & Chow, T. W. (02 de 1999). Adaptive regularization parameter selection method for enhancing generalization capability of neural networks. *Artificial Intelligence, 107*(2), 347-356.

Makridakis, S., Wheelwright, S., & Hyndman, R. (1998). *Forecasting: Methods and applications.* New York: John Wiley & Sons.

Makridrakis, S. G., Wheelwright, S. C., & Hyndman, R. J. (1998). *Forecasting: Methods and Applications.* New York: John Wiley & Sons.

Marquardt, D. W., & Snee, R. D. (Feb. de 1975). Ridge regression in practice. *The American Statistician, 29*(1), 3-20.

Masters, T. (1993). *Practical neural network recipes in C++.* New York: Academic Press.

Mishra, S., & Patra, S. (2009). Short term load forecasting using a novel recurrent neural network. *International Journal of Computational Intelligence: Theory and Practice, 4*(1), 39-45.

Montgomery, D. C., Johnson, L. A., & Gardiner, J. S. (1990). *Forecasting & Time Series Analysis* (Segunda ed.). Singapure: McGraw-Hill, Inc.

Montgomery, D., & Jenni, C. (2015). *Introduction to Time Series Analysis and Forecasting* (Segunda ed.). NY: Wiley.

Moody, J. E., & Rögnvaldsson, T. (1997). Smoothing regularizers for projective basis function networks. *Advances in Neural Information Processing Systems*, 585 - 591.

Morozov, V. (1984). *Methods for Solving Incorrectly Posed Problems.* Berlin: Springer-Verlag.

Morozov, V. (1993). *Regularization Methods for Ill-Posed Problems.* Boca Raton, FL: CRC Press.

Morozov, V. A. (1966). On the solution of functional equations by the method of regularization. *Soviet Math*, 414-417.

Nowlan, S. J., & Hinton, G. E. (1992). Simplifying neural networks by soft weight-sharing. En J. Moody, S. Hanson, & R. Lippmann (Edits.), *Advances in Neural Information Processing Systems* (Vol. 4, págs. 173-193). California, USA: Morgan Kaufmann, San Mateo.

Ortíz, D. M., Villa, F. A., & Velásquez, J. D. (2007). Una Comparación entre Estrategias Evolutivas y RPROP para la Estimación de Redes Neuronales. *Avances en Sistemas e Informática, 4*(2), 135–144.

Palit, A. K., & Popovic, D. (2005). *Computational Intelligence in Time Series Forecasting.* London: Springer.

Parlos, A., Rais, O., & Atiya, A. (2000). Multi-step-ahead prediction using dynamic recurrent neural networks. *Neural Networks, 13*, 765-786.

Peña, D. (1994). Second-generation time-series models: a comment on ‘Some advances in non-linear and adaptive modelling in time-series analysis’ by Tiao and Tsay. *Journal of Forecasting, 13*, 133-140.

Rao, T. S., & Gabr, M. (1984). An introduction to bispectral analysis and bilinear time series models. *Lecture Notes in Statistics, 24*, 528–535.

Rast, M. (1997). Forecasting Financial Time Series with Fuzzy Neural Networks. *IEEE lnternational Conference on Intelligent Processing Systems*, págs. 432-434.

Reginska, T. (1996). A regularization parameter in discrete ill-posed problems. *SIAM J. Sci. Comput, 17*(3), 740–749.

Riedmiller, M. (1994). Advanced supervised learning in multi-layer perceptrons – from backpropagation to adaptive learning algorithms. *Computer Standards and Interfaces, 16* , 265–278.

Riedmiller, M., & Braun, H. (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. En *Proceedings of the IEEE International Conference on Neural Networks* (págs. 586–591). IEEE Press.

Sánchez, P. A., & Velásquez, J. D. (2011). El rol del algoritmo de entrenamiento en la selección de modelos de redes neuronales. *Rev. U.D.C.A Act. & Div. Cient., 1*(14), 149 - 156.

Sánchez, P., & Velásquez, J. D. (2010). Problemas de Investigación en la Predicción de Series de Tiempo con Redes Neuronales Artificiales. *Revista Avances en Sistemas e Informática, 7*(3), 67-73.

Sarle, W. (1994). The 19th Annual SAS Users Group Int. Conference. *Neural networks and statistical models* (págs. 1538–1550). Cary, North Carolina: SAS Institute.

Schittenkopf, C., Deco, G., & Brauer, W. (1997). Two strategies to avoid overfitting in feedforward networks. *Neural Networks, 10*(3), 505-516.

Scott, A. B. (2012). *An Introduction to Enterprise Architecture.* Bloomington, IN: Author House.

Setiono, R. (1997). A penalty-function approach for pruning feedforward neural networks. *Neural Computation, 9*(1), 185-204 .

Shen, H. (2014). Interactive Notebooks: Sharing The Code. *Nature, 515*, 151-152.

Suykens, J. A., Van Gestel, T., Brabanter, J., Moor, B., & Vandewalle, J. (2005). *Least Squares Support Vector Machines.* K. U. Leuven, Belgium: World Scientific.

Tetko, I., & Villa, A. (1997). An Enhancement of Generalization Ability in Cascade Correlation Algorithm by Avoidance of Overfitting/Overtraining Problem. *Neural Processing Letters, 6*(1), 43-50 .

Tikhonov, A. (1963). On Solving Incorrectly Posed Problems and Method of Regularization. *Docklady Academii Nauk, 151*, 501-504.

Tikhonov, A. (1973). On Regularization of ill-posed problems. *Docklady Academii Nauk, 153*, 49-52.

Tikhonov, A., & Arsenin, V. (1977). *Solutions of Ill-Posed problems.* Washington, DC: W.H. Winston.

Tjostheim, D. (1994). Nonlinear time series: a selective review. *Scand. J. Statist., 21*, 97-130.

Tong, H. (1990). *Nonlinear Time Series: A Dynamical System Approach.* Oxford: Oxford University Press.

Tong, H. (2011). Threshold models in time series analysis —30 years on (with discussions by P.Whittle, M.Rosenblatt, B.E.Hansen, P.Brockwell, N.I.Samia & F.Battaglia). *Statistics & Its Interface*(4), 107-136.

Tong, H., & Lim, K. (1980). Threshold autoregressive, limit cycles and cyclical data. *Journal of the Royal Statistical Society Series B, 42 (3). Pág. 245–292, 42*(3), 245–292.

Vapnik, V. (1998). *Statistical Learning Theory.* New York: Wiley.

Velásquez, J. D., & Villa, F. A. (2008). Una comparación entre perceptrones multicapa y redes cascada correlacion para el pronostico de series de tiempo. En C. M.

Zapata, & G. L. Giraldo (Edits.), *Tendencias en Ingeniería de Software e Inteligencia Artificial* (Vol. 2, págs. 67-74). Medellín, Colombia: LitoNueve.

Velásquez, J. D., Dyner, R., & Souza, R. C. (2005). Predicción Condicional del Precio Mensual de Bolsa basada en Escenarios de Eventos Hidrológicos Extremos. *VII Seminario Internacional sobre Análisis y Mercados Energéticos & I Seminario CERES.* Bogotá, Colombia.

Villa, F. A., & Velásquez, J. D. (2010). *Regularización de Redes Cascada Correlación con Regresión en Cadena.* Cartagena: Quinto Congreso Colombiano de Computación (Abril 14 – 16).

Villa, F. A., Velásquez, J. D., & Souza, R. C. (2008). Una aproximación a la regularización de redes cascada-correlación para la predicción de series de tiempo. *Investigación Operacional.*(28), 151-161.

Wahba, G. (1990). Spline Models for Observational Data. *CBMS-NFS Regional Conference Series in Applied Mathematics. 59.* Philadelphia: SIAM.

Wei, W. W. (2006). *Time Series Analysis : Univariate and Multivariate Methods* (Segunda ed.). USA: Addison-Wesley.

Weigend, A. S., Rumelhart, D. E., & Huberman, B. A. (1991). Generalization by weight-elimination with application to forecasting. En R. P. Lippmann, J. E. Moody, & D. S. Touretzky (Edits.), *Advances in Neural Information Processing Systems* (ISBN:1-55860-184-8 ed., Vol. 3, págs. 875–882). San Mateo, CA, USA: Morgan Kaufmann Publishers Inc.

Xu, Q., & Nakayama, K. (1997). Avoiding Weight-illgrowth: Cascade Correlation Algorithm with Local Regularization. *Neural Networks, 3*, 1954 - 1959.

Yan, X.-B., Wang, Z., Yu, S.-H., & Li, Y.-J. (2005). Time Series Forecasting with RBF Neural Network. *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, 4680-4683.

Yong Castillo, E. (2003). *Competencias Comunicativas y Aprendizaje Autónomo: Guía N° 1.* Bogotá.

Zhang, D., Han, Y., Ning, X., & Liu, .. (2008). A Framework for Time Series Forecasts. *Proceedings ISECS International Colloquium on Computing, Communication, Control, and Management, 1*, 52-56.

Zhang, G. (2003). Time Series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing, 50*, 159–175.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (Marzo de 1998). Forecasting with artificial neural networks: the state of the art. *International Journal of Forecasting, 14*(1), 35-62.

Zhang, P., Patuwo, B., & Hu, M. (2001). A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research, 28*(4), 381-396.