

Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA

Antonino Parisi F.

Universidad de Chile

Extracto

Este estudio analiza la capacidad de las redes neuronales para predecir el signo de las variaciones semanales del IPSA. De las diversas arquitecturas de redes neuronales utilizadas sobre el periodo comprendido entre el 11 de Enero de 1999 y el 22 de Octubre de 2001, la Red Ward Recursiva obtuvo el mejor rendimiento, alcanzando una capacidad predictiva extramuestral de 72% y una rentabilidad acumulada extramuestral para la cartera conformada por el IPSA de un 24,42%; la Red Recurrente de Jordan y Elman Recursiva obtuvo una capacidad predictiva de un 64% y una rentabilidad de 21.33%; mientras que el modelo ingenuo generó un retorno de 18,31%, mayor al de la Red Ward Estándar y el MLP Recursivo, a pesar de no evidenciar capacidad predictiva, lo que permitiría concluir que una mayor capacidad predictiva no siempre se traduce en mayores retornos. El test de Pesaran y Timmermann (1992) arroja evidencia de que, para el caso chileno, la Red Ward Recursiva y la Red Jordan y Elman, en su versión

El autor agradece a Franco Parisi por sus comentarios y a Edinson Cornejo por su labor como asistente de investigación. También agradezco los comentarios realizados por José Luis Guerrero, John Cortner y un árbitro anónimo.

recursiva y estándar, pueden predecir el cambio direccional del índice. A su vez, estos modelos podrían producir mayores retornos que un modelo ingenuo. Lo anterior apoya las conclusiones del estudio de Leung, Daouk y Chen (2000), según el cual la predicción de la dirección del movimiento puede arrojar mayores ganancias de capital que la proyección del valor de cierre.

CJEL: G10, G14 y G15

Conceptos Claves: Redes Neuronales, Perceptrón Multicapa, Test Directional Accuracy, Capacidad Predictiva.

Abstract

This paper analyzes the capacity of the neural networks to forecast the sign of weekly variations of IPSA. Several architectures of neural networks were used over the time period between January 11th of 1999 to October 22th of 2001, being the Recursive Ward Network the one with the best performance, reaching an outsample predictive capacity of 72% and an outsample accumulate yield for the IPSA portfolio of a 24.42%. The Recurrent Recursive Jordan & Elman Network achieved a forecast ability of 64% and a return of 21.33%; while the AR(1,1) model obtained a return of 18.31%, higher than the Ward Standard Network and Recursive MLP returns. Even though the first one had not statistical evidence of predictive capacity it would allow to conclude that a higher predictive capacity do not always implies higher yields. The Pesaran & Timmermann test (1992) gives evidence that the Recursive Ward Network and the Recurrent Recursive Jordan & Elman Network, in his recursive and standard version can forecast the index directional change for the chilean case. Also, this models could produce higher returns than AR(1,1) model. This result supports the conclusions of Leung, Daouk & Chen (2000) about the prediction of movement directions can give greater capital gains that the forecasting of close values.

1. Introducción

Este estudio analiza la capacidad de los modelos de redes neuronales para predecir el signo de la rentabilidad o variación semanal del índice de precios selectivo de acciones (IPSA) del mercado bursátil chileno, entendiendo que la predicción de la dirección del movimiento del índice del mercado accionario es relevante para desarrollar estrategias de transacción efectivas, las cuales pueden arrojar mejores resultados que aquellas basadas en la proyección del valor de la variable observada (Leung, Daouk y Chen, 2000). En este contexto, la contribución de este estudio radica en el análisis de diversas arquitecturas de redes neuronales a fin de medir su capacidad para predecir el movimiento del IPSA, sobre la base del número de veces en que la dirección pronosticada es correcta, para finalmente determinar cuál de ellas es la más apropiada. Además, aún cuando existen estudios que muestran la fortaleza de las redes neuronales en la predicción de series de tiempo no lineales con respecto a los métodos estadísticos tradicionales, se requieren estudios adicionales para evaluar la aplicación de las redes neuronales a la proyección y toma de decisiones, con el objeto de hacerlas instrumentos confiables para los analistas encargados de realizar pronósticos.

Nuestro análisis se centrará en el empleo de las redes neuronales multicapas con aprendizaje supervisado, específicamente el Perceptrón Multicapa y algunas de sus variantes. Se utilizarán estos modelos de redes ya que actúan como funciones mapeadoras universales, desempeñándose muy bien con las series de tiempo. A su vez, se emplearán distintas arquitecturas de redes neuronales y se evaluará la capacidad predictiva de cada una de ellas a través de la prueba de signo.

El documento se divide en cinco secciones: la sección dos presenta una revisión de la literatura relacionada a este estudio, la sección tres explica la metodología empleada en la investigación; la sección cuatro aborda la simulación y el análisis de los resultados y, finalmente, la sección cinco presenta las conclusiones del estudio.

2. Revisión de la literatura

Lo y MacKinlay (1988), empleando datos de mercados bursátiles desarrollados tales como U.S.A., Europa Occidental y Japón, concluyen que existe evidencia significativa de que los precios accionarios no siguen un camino aleatorio y muestran que los retornos accionarios son predecibles en algún grado. Lo anterior se contrapone con la hipótesis de mercados eficientes (Fama, 1970), donde la eficiencia significa que el mercado refleja completa y correctamente toda la información relevante para la determinación de los precios de los activos. Esta hipótesis, por lo general, no es aceptada y, en consecuencia, se ha intentado emplear modelos no lineales para mejorar el ajuste y la predicción de la rentabilidad de los activos bursátiles. Al hallazgo de Lo y MacKinlay (1988), quienes registran una correlación serial positiva entre los retornos semanales, se añaden los resultados de DeBondt y Thaler (1985), Fama y French (1988), Poterba y Summers (1988) y Chopra, Lakonishok y Ritter (1992), quienes encuentran una correlación serial negativa en los retornos de los activos individuales y varios portfolios, sobre intervalos de tres a diez años. En este contexto, Jegadeesh (1990) examina la predictibilidad de los retornos mensuales sobre activos individuales y encuentra una correlación serial negativa de primer orden altamente significativa para rezagos de dos meses y una correlación serial positiva para rezagos mayores.

Al momento de explicar la predictibilidad de las variaciones de los retornos accionarios se postulan dos argumentos alternativos: (a) los mercados son ineficientes y los precios de los activos se mueven alrededor de su valor fundamental, y (b) los mercados son eficientes y la predictibilidad de las variaciones puede ser explicada por un equilibrio en los retornos *time-varying*. Al respecto, Ferson y Harvey (1991) muestran que la predictibilidad de los retornos accionarios no se debe necesariamente a ineficiencias del mercado o a una sobrerreacción de los inversionistas irracionales, sino que a la predictibilidad que presentan algunas variables agregadas que son parte del *set* de información que explica la rentabilidad de los activos. En este sentido, un estudio de eventos de transacciones de *insiders* de Ferson y Schadt (1996), muestra que la omisión de variables tales como rezagos del retorno accionario y de la tasa de interés podrían

conducir a resultados erróneos en la proyección. Además, según Leung, Daouk y Chen (2000), la predicción de los retornos accionarios, dadas las variables agregadas en el *set* de información de los inversores, es un hecho bien aceptado en la reciente literatura de las finanzas empíricas, y las preguntas apuntan hacia cómo usar la información de una manera óptima para predecir y transar en los mercados, cuestión que es relevante ya que una estimación puntual relativamente exacta puede no ser un buen predictor de la dirección del cambio que experimentará el nivel de precios del instrumento.

Estudios recientes han sugerido que las estrategias de transacción basadas en proyecciones de la dirección del cambio en el nivel de precios son más efectivas y pueden generar beneficios más altos que aquellas basadas en una predicción puntual del nivel de precios de los instrumentos financieros. En esta línea de estudio, Wu y Zhang (1997) investigaron la predictibilidad de la dirección del movimiento en el tipo de cambio *spot* futuro. Aggarwal y Demaskey (1997) encontraron que el desempeño de la cobertura cruzada mejora significativamente si se puede predecir la dirección de las variaciones en el tipo de cambio. Maberly (1986) analizó la relación entre la dirección de los cambios de precio interdía e intradía, y O'Connor, Remus y Griggs (1997) concluyeron que los individuos muestran diferentes tendencias y comportamientos para series que se mueven al alza y a la baja, junto con demostrar la utilidad de proyectar la dirección del cambio en el nivel de precios.

No obstante, el grado de predictibilidad de los retornos es generalmente considerado económicamente no significativo, por lo que Hodgson y Nicholls (1991) sugieren evaluar la significancia económica de predecir la dirección de los cambios en los precios de los activos. En esta materia, Leung, Daouk y Chen (2000) compararon la capacidad predictiva de los modelos de clasificación (*linear discriminant analysis*, *logit model*, *probit model* y *probabilistic neural network*) con los de estimación de nivel (*adaptive exponential smoothing*, *vector autoregression with Kalman filter*, *multivariate transfer function* y *multilayered feedforward neural network*) y concluyeron que los primeros (los cuales son propuestos para predecir el signo o la dirección del retorno de los índices bursátiles S&P 500, FTSE 100 y Nikkei 225) se desempeñan mejor que los segundos en términos de

su tasa de acierto, medida por el número de veces en que la dirección pronosticada es correcta y, además, son capaces de generar beneficios más altos. Esto último resulta relevante para los analistas de mercado y para los *traders*, en el sentido de que apunta a que deben centrar sus esfuerzos en predecir con precisión la dirección de los movimientos en vez de minimizar la desviación de las estimaciones de los valores observados.

A. Las redes neuronales

De acuerdo a Martín del Brío y Sanz (1997), las redes neuronales artificiales¹ (R.N.A.) “son sistemas de procesamiento que copian esquemáticamente la estructura neuronal del cerebro para tratar de reproducir sus capacidades”. En consecuencia, son una clase de modelos no lineales flexibles que se caracterizan por ser sistemas paralelos (cuentan con una gran cantidad de neuronas o procesadores elementales, PEs, cada uno de los cuales trabaja paralelamente con una pequeña parte de un problema mayor), distribuidos (cuentan con muchos PEs a través de los cuales distribuyen su memoria) y adaptativos (tienen la capacidad de adaptarse al entorno modificando sus pesos y sinapsis de manera de encontrar una solución aceptable al problema). Según estos autores, estos tres conceptos se traducen en un mejor rendimiento y en una mayor velocidad de procesamiento. Por su parte, Herbrich, Keilbach, Graepel, Bollmann-Sdorra y Obermayer (2000) señalan que la característica más importante de las redes neuronales es su capacidad para aprender dependencias basadas en un número finito de observaciones, donde el término aprendizaje significa que el conocimiento adquirido a partir de las muestras puede ser generalizado a las observaciones no vistas, es decir,

¹Las redes neuronales pueden entenderse como modelos multicuacionales o multi-etapas, en los que el *output* de unos constituye el *input* de otros. En el caso de las redes multicapas, existen etapas en las cuales las ecuaciones operan en forma paralela. Los modelos de redes neuronales, al igual que, por ejemplo, los modelos de suavizamiento exponencial y de análisis de regresión, utilizan *inputs* para generar un *output* en la forma de una proyección. La diferencia radica en que las redes neuronales incorporan inteligencia artificial en el proceso que conecta los *inputs* con los *outputs* (Kuo y Reitsch, winter 1995-96).

puede ser empleado para proporcionar una respuesta correcta ante datos no utilizados en el entrenamiento de la red. La literatura sugiere que las redes neuronales poseen varias ventajas potenciales sobre los métodos estadísticos tradicionales, destacándose el que éstas pueden ser aproximadores de funciones universales aún para funciones no lineales (Hornik, Stinchcombe y White, 1989), lo que significa que ellas pueden aproximar automáticamente cualquier forma funcional (lineal o no lineal) que mejor caracterice los datos, permitiéndole a la red extraer más señales a partir de formas funcionales subyacentes complejas (Hill, Márquez, O'Connor y Remus, 1994).

Los PE son un dispositivo simple de cálculo que, a partir de un vector de entrada procedente del exterior o de otras neuronas, proporciona una única respuesta o salida, la que se puede conectar a otros PE de la red. Los PE se agrupan formando capas las que, a su vez, conforman las llamadas redes neuronales (Freeman y Skapura, 1993). Los componentes de una red neuronal son: (a) unidades de entrada, las cuales reciben información proveniente de otras neuronas o desde el exterior, para luego traspasarla a la capa siguiente sin realizar procesamiento alguno; (b) unidades de salida, las cuales corresponden al resultado de la red neuronal; y (c) unidades ocultas, que son aquellas en donde se lleva a cabo el procesamiento de los datos. Luego, el problema se centra en determinar los parámetros adecuados de las redes neuronales que contengan la combinación correcta de elementos de procesamiento, tasas de aprendizaje, arquitectura y número de capas con tiempos de entrenamientos aceptables y que posean un buen rendimiento.

B. Aplicaciones de los modelos de redes neuronales

Algunos investigadores han encontrado que, en general, los mercados financieros se comportan de una forma no lineal (Bosarge, 1993), cuestión que ha favorecido el empleo de modelos de redes neuronales ya que éstas han evidenciado un buen desempeño en modelamientos no lineales (Widrow, Rumelhart y Lehr, 1994).

Es posible distinguir al menos dos importantes aplicaciones de las redes neuronales en las áreas de economía y finanzas: primero, la clasificación de agentes económicos tales como compañías, para obtener una estimación de la probabilidad de quiebra (por ejemplo, Odom y Sharda, 1990; Wilson y Sharda, 1994; Rahimian, 1993), compañías de seguros (Brockett, Cooper, Golden y Pitaktong, 1994), la capacidad acreedora de clientes bancarios (Marose, 1990), tenedores de tarjetas de crédito (Jagielska y Jaworski, 1996), etc.; segundo, la predicción de series de tiempo (Tang, 1991, et al). Con respecto a esto último, cabe señalar que el propósito de un modelo de predicción es capturar patrones de comportamiento en datos multivariados que distingan varios resultados, cosa que es bien realizada por los modelos no paramétricos de redes neuronales (Gorr, 1994). Los modelos de redes neuronales han sido desarrollados para predecir valores de índices bursátiles y de activos individuales, situándose la mayoría de las primeras investigaciones y aplicaciones en mercados establecidos en U.S.A. (Bosarge, 1993; Tsibouris y Zeidenberg, 1995; White, 1993), Gran Bretaña (Tsibouris y Zeidenberg, 1995) y Japón (Yoda, 1994). Dichos modelos han sido empleados para predecir el nivel o el signo de los retornos de índices bursátiles, entre otras aplicaciones relacionadas a la toma de decisiones en las áreas de finanzas e inversión (Hawley, Johnson y Raina, 1990; Refenes, 1995).

Sin embargo, para una clara comprensión y una correcta utilización de los modelos de redes neuronales, se debe tener presente las limitaciones presentadas por ellos: sus modelos estimados son difíciles de explicar debido a su compleja forma funcional; los modelos de redes neuronales no lineales no poseen propiedades estadísticas clásicas, razón por la cual no es posible realizar pruebas de hipótesis y construir intervalos de confianza; puede ocurrir un sobreajuste a los datos de la muestra o un sobreaprendizaje de la red (el cual se produce cuando el sistema se ajusta demasiado a los datos de entrenamiento, aprendiendo incluso el ruido presente en ellos, por lo que crece el error ante patrones diferentes a los empleados en el entrenamiento y disminuye la precisión de la proyección), dados los muchos parámetros que necesitan ser estimados para los modelos multivariados (Gorr, 1994); y no existe una guía objetiva para seleccionar la dimensión apropiada de la red (es decir, el número de capas ocultas o neuronas), donde la implementación

usualmente se realiza a través de un proceso de prueba y error, aunque se ha sugerido el empleo de métodos sistemáticos tales como el *Support Vector Networks Method* (Herbrich, Keilbach, Graepel, Bollmann-Sdorra y Obermayer, 2000).

3. Metodologías y datos

Para esta investigación se emplearon 143 valores de cierre semanales de las variables incluidas en el modelo, correspondientes al período comprendido entre el 11 de Enero de 1999 y el 22 de Octubre de 2001. En el modelo formulado, la variable de salida está dada por la variación porcentual o rentabilidad del IPSA correspondiente al período "t" ($\delta IPSA_t$), mientras que las variables de entrada se referirán a las variaciones porcentuales del tipo de cambio y de los índices bursátiles Dow Jones de Industrias, Bovespa, Nikkei e IPSA, correspondientes al período anterior, es decir, al período "t-1" (δTC_{t-1} , δDJI_{t-1} , $\delta Adrian_{t-1}$, $\delta Bovespa_{t-1}$, $\delta Nikkei_{t-1}$ y $\delta IPSA_{t-1}$, respectivamente, ver cuadro 1). La forma funcional del modelo se expresa en la ecuación (1):

$$\delta IPSA_t = g(\delta TC_{t-1}, \delta DJI_{t-1}, \delta Bovespa_{t-1}, \delta Nikkei_{t-1}, \delta IPSA_{t-1}) \quad (1)$$

Cuadro 1

IPSA _t	IPSA _{t-1}	TC _{t-1}	DJI _{t-1}	ADRIAN _{t-1}	BOVESPA _{t-1}	NIKKEI _{t-1}
-0.043	*	*	*	*	*	*
0.061	-0.043	0.008	0.018	-0.038	0.072	-0.010
-0.062	0.061	-0.004	-0.010	0.021	-0.059	-0.025
-0.022	-0.062	0.008	-0.003	-0.040	-0.020	-0.039
0.039	-0.022	-0.003	0.018	0.002	0.042	0.024
-0.031	0.039	0.007	0.003	0.011	-0.161	-0.013
0.106	-0.031	-0.001	0.009	-0.045	-0.086	-0.015
-0.031	0.106	-0.005	0.035	0.061	0.082	-0.030
0.009	-0.031	0.006	-0.004	-0.016	-0.086	-0.025
*	0.009	-0.003	0.050	0.000	-0.101	-0.040

La razón por la que se utilizaron variables rezagadas se debe a que no se cuenta con capacidad para predecir las variables independientes utilizadas en el modelo (tipo de cambio, Dow Jones, etc.). Además, al utilizar variables rezagadas se está considerando el desfase lógico que existe entre el momento en que se realiza la predicción, el período actual o (t-1), y el momento futuro para el cual ésta es válida, al que nos referimos como período (t). Así, debido a que la información de hoy está desfasada con respecto a la proyección, se utilizan modelos dinámicos con un rezago (t-1) en las variables independientes.

Por otro lado, el *software* a utilizar es el Neuroshell 2, el cual requiere para su funcionamiento que la base de datos se divida en tres conjuntos diferentes: (a) conjunto de entrenamiento² (porcentaje de datos empleados para que la red aprenda el problema), (b) conjunto de prueba (conjunto de datos que se utilizará para evitar el sobreaprendizaje de la red y que, unido a las observaciones del conjunto de entrenamiento, se extiende desde el 11 de Enero de 1999 hasta el 30 de Octubre de 2000), y (c) conjunto de producción o extramuestral (datos que no han sido incorporados anteriormente, que serán utilizados para probar el rendimiento o la capacidad de predicción de la red ante datos que nunca ha visto, y que pertenecen al subperíodo que abarca desde el 06 de Noviembre de 2000 hasta el 22 de Octubre de 2001). Cabe mencionar que otros autores dividen el período de datos en dos segmentos, donde el primero es utilizado para entrenar la red mientras que el segundo es usado para determinar y validar su arquitectura y la especificación del modelo (Chen y Leung, 1998; Leung, Daouk y Chen, 2000).

²En una red neuronal hay pares de *inputs* y *outputs* que son usados para entrenar la red. Pueden haber múltiples *inputs* (variables explicativas) y múltiples *outputs* (proyecciones de diferentes variables). Entre los *inputs* y los *outputs* hay una capa (o múltiples capas) de procesamiento que imitan el trabajo del cerebro humano. Luego, dado un nuevo conjunto de *inputs*, la red puede producir un nuevo *output* (proyección) sobre la base de lo que aprendió de los pares de *inputs* y *outputs* que le fueron provistos. El analista puede controlar algunos aspectos del proceso, tales como la tasa de aprendizaje y la precisión deseada del *output* (Wilson y Keating, 1998).

A. Tipo de red neuronal y función de activación

Para abordar esta tarea se emplearon diversos modelos de redes neuronales que usan el algoritmo de aprendizaje supervisado de propagación hacia atrás, los cuales permitirán predecir el signo de la rentabilidad del IPSA. Estos modelos se caracterizan porque en ellos el resultado es conocido y la red se entrena a sí misma hasta que es capaz de predecir aquel resultado asociado con los datos de entrada (Dasgupta, Dispensa y Ghose, 1994). Los modelos analizados son: el Perceptrón Multicapa Estándar, la Red Recurrente Jordan-Elman y la Red Ward.

a. EL PERCEPTRÓN MULTICAPA ESTÁNDAR

El Perceptrón Multicapa (MLP), descrito formalmente por Werbos (1974) y posteriormente por Parker (1985) y por McClelland y Rumelhart (1986), es una red neuronal unidireccional entrenada con el algoritmo de propagación hacia atrás (Rumelhart, Hinton y Williams, 1986) y su modo de aprendizaje es supervisado: el retorno del índice bursátil es usado para supervisar el proceso de entrenamiento y, a la vez, actúa como una variable explicativa ya que éste es el resultado deseado (en otras palabras, para predecir la variación futura del índice se requiere información relativa a su variación presente).

Los parámetros del algoritmo de propagación hacia atrás, así como la topología de la red, necesitan ser ajustados para un desempeño óptimo, por lo que se debe experimentar hasta encontrar los parámetros correctos para confeccionar la red que se utilizará en la realización de los pronósticos. La red aprende un conjunto predefinido de pares de entradas y salidas dadas como ejemplos, empleando un ciclo propagación-adaptación de dos fases: una vez que se han aplicado los datos de entrada a la primera capa de unidades de la red, ésta se va propagando a través de todas las capas superiores hasta generar una salida, la cual se compara con la salida deseada para estimar el error asociado a la predicción. El error se transmite hacia atrás, desde la capa de salida hacia todas las capas intermedias que contribuyan directamente a ella. Sin embargo, las unidades de la capa

intermedia sólo reciben una fracción del error total, basándose en la contribución relativa de la unidad a la salida original. Este proceso se repite, capa por capa. Luego, basándose en el error, se actualizan los pesos de conexión de cada unidad con el objeto de ajustar el modelo y disminuir el error final. Así, a medida que se entrena la red, las neuronas de las capas intermedias se organizan a sí mismas de tal modo que ellas aprenden a reconocer distintas características del conjunto de entrenamiento, aproximándose de esta manera al pensamiento y al comportamiento de los agentes, en nuestro caso, del mercado bursátil.

El proceso mediante el cual el Perceptrón Multicapa estándar obtiene la salida (predicción) y actualiza los pesos de conexión entre las capas, junto con la aplicación del algoritmo de aprendizaje de retropropagación del error (de acuerdo a Freeman y Skapura [1993]), se presenta en el anexo 1. La implementación de este procedimiento una y otra vez generará un proceso de aprendizaje que le permitirá a la red adecuar sus pesos para reducir el error de predicción. Según Martín del Brío y Sanz (1997), el aprendizaje se puede dar por concluido cuando el error resulta aceptablemente pequeño para todos los pares de vectores de entrenamiento, es decir, cuando se logra minimizar el error de la estimación. Cabe señalar que el término de error que emplean los *softwares* de redes neuronales es el error cuadrático medio. No obstante, la medida de error empleada en este estudio estará dada por la prueba de signo ya que, como se mencionó anteriormente y de acuerdo a Leung, Daouk y Chen (2000), los modelos propuestos para predecir el signo o la dirección del retorno de los índices bursátiles se desempeñan mejor que los modelos de proyección de nivel en términos de su tasa de acierto (medida por el número de veces en que la dirección pronosticada es correcta) y, además, son capaces de generar beneficios más altos.

b. LA RED RECURRENTE JORDAN-ELMAN

Por otra parte, el MLP con conexiones recurrentes es conocido por su habilidad para aprender secuencias y, por lo tanto, es recomendable para series de tiempo. Las redes con algoritmo de propagación hacia atrás y

conexiones recurrentes son entrenadas de la misma forma, excepto que en esta última los patrones deben ser presentados siempre en el mismo orden y la selección aleatoria no está permitida, por el hecho de operar con series de tiempo. La única diferencia en estructura es que existe un *slab* (grupo de neuronas del mismo tipo) extra³ en la capa de entrada que está conectado a la capa oculta, al igual que el otro *slab* de entrada, lo que le permite a la red ver el conocimiento anterior que tenía sobre las entradas previas y, de esta manera, operar con rezagos de las variables explicativas. Existen 3 tipos de redes MLP con conexiones recurrentes: (a) la capa de entrada realimentada en la capa de entrada; (b) la capa oculta realimentada en la capa de entrada; (c) la capa de salida realimentada en la capa de entrada. Se recomienda usar la arquitectura en donde la capa oculta es realimentada en la capa de entrada, lo que significa que las características detectadas en todos los patrones anteriores son alimentadas a la red con cada nuevo patrón, convirtiéndola en la red recurrente más poderosa. Esta arquitectura es comúnmente conocida como la Red Recurrente de Jordan-Elman.

c. LA RED WARD

Finalmente, la Red Ward se caracteriza porque pueden aplicarse diferentes funciones de activación a los *slab* de la capa oculta, para detectar diferentes características en los patrones procesados a través de la red. Por ejemplo, un diseño de una red puede usar una función de activación gaussiana en un *slab* oculto para detectar diferentes características en el rango medio de los datos, y usar un complemento gaussiano en otro *slab* oculto para detectar características de los extremos superiores e inferiores de los datos. De esta manera, la capa de salida tendrá diferentes puntos de vista de los datos, lo cual puede conducir a una mejor predicción. Las arquitecturas de redes neuronales empleadas en este estudio se presentan en el anexo 2.

³Este *slab* extra es algunas veces llamado el "término largo" de memoria de la red.

B. Evaluación de la capacidad predictiva de la red neuronal y análisis comparativo

Cada uno de los modelos de proyección descritos anteriormente fue estimado y validado a partir de los datos del conjunto de entrenamiento. Este proceso fue seguido por una evaluación empírica sobre la base de los datos del conjunto extramuestral⁴, el cual se compone de 50 observaciones semanales. En esta etapa, el desempeño relativo de los modelos fue medido por el número de predicciones correctas (*hits*) del signo del retorno del índice. Así, al igual que Kanas (2001), la proyección extramuestral fue testada en función de su precisión direccional (*directional accuracy*, DA), a través de la prueba de signo. Para examinar la precisión direccional se empleó el test de Pesaran y Timmermann (1992), el cual compara el signo de la proyección, $\hat{y}_{n+i}$, con el del valor observado, y_{n+i} , para cada *i*-ésima observación del conjunto extramuestral ($i=1,2, \dots, m$), donde el signo indica la dirección en que se moverá el mercado accionario: al alza, si es positivo, o a la baja, si es negativo. Si los signos coinciden, aumenta la efectividad de la red, y en caso de no existir coincidencia, aumenta el error de predicción de la red. Este proceso se realizó para cada observación del conjunto extramuestral a través del método directo⁵ de proyección multi-etapa, por lo que no se evaluó la predicción del conjunto extramuestral en su totalidad de una sola vez, sino que se fue analizando una a una como parte de un proceso recursivo, metodología que ha sido empleada para medir el desempeño de modelos de redes neuronales que buscan predecir períodos de recesión en los Estados Unidos (Qi, 2001; Estrella y Mishkin, 1998). De esta forma, si consideramos que el conjunto extramuestral tiene "m" datos, al evaluar el funcionamiento de la red se consideró sólo la predicción del primer valor. Posteriormente, el dato analizado salió del conjunto extramuestral

⁴Cabe mencionar que Sharda (1994) señala que los resultados basados en tal comparación están sujetos a una muestra sesgada, lo que atenta contra su significancia estadística.

⁵Con el método directo, el error de proyección no es usualmente un ruido blanco ya que puede tener relaciones temporales. Sin embargo, esto no presenta un problema en el presente estudio ya que no se hacen inferencias estadísticas basadas en el error de proyección.

(quedando con “m-1” datos) y pasó a formar parte del conjunto de entrenamiento, siguiendo un proceso aleatorio, por lo que la muestra de “n” datos que contiene los valores de entrada se incrementó a “n+1”. Luego se realizó una nueva iteración, lo que implicó construir un nuevo modelo para cada una de las “m-1” proyecciones, permitiéndole a la red aprender del error cometido en la predicción y ajustar nuevamente los pesos estimados para los valores de entrada de las capas de salida y oculta, w_{jk} y w_{ij} , respectivamente. Este proceso se repitió hasta que en el conjunto extramuestral quedó sólo un dato, por lo que la red recalculó los pesos w_{jk} y w_{ij} hasta el momento en que la última observación (correspondiente al período “t-1”) es empleada para proyectar el valor que la variable de salida podría alcanzar en el momento “t”, el cual representa el futuro inmediato (ver cuadro 2).

Cuadro 2
Ejemplo prueba de signo

PERÍODO	OBSERVADO	PREDICCIÓN	PRUEBA
1	0.068157	0.043550	1
2	-0.003956	0.002210	
3	0.032541	0.031627	
4	-0.030830	0.028130	

El ratio de éxito (SR) se define como:

$$SR = m^{-1} \sum_{i=1}^m I_i [y_{n+i}, \hat{y}_{n+i} > 0] \quad (2)$$

donde $I_i[\cdot]$ es una función indicador que toma el valor de 1 cuando su argumento es cierto y 0 en otro caso. Además:

$$P = m^{-1} \sum_{i=1}^m I_i [y_{n+i} > 0] \quad (3)$$

y

$$\hat{P} = m^{-1} \sum_{i=1}^m I_i [\hat{y}_{n+i} > 0] \quad (4)$$

El ratio de éxito en el caso de independencia de $\hat{y}_{n+i}$ y y_{n+i} , SRI, está dada por:

$$SRI = P \cdot \hat{P} + (1 - P)(1 - \hat{P}) \quad (5)$$

cuya varianza es:

$$VAR[SRI] = m^{-2} [m(2 \cdot \hat{P} - 1)^2 P(1 - P) + m(2 \cdot P - 1)^2 \hat{P}(1 - \hat{P}) + 4 \cdot P \cdot \hat{P}(1 - P)(1 - \hat{P})] \quad (6)$$

Por su parte, la varianza de SR se define como:

$$VAR[SR] = m^{-1} \cdot SRI(1 - SRI) \quad (7)$$

Finalmente, el test DA de Pesaran y Timmermann (1992) está dado por:

$$DA = (VAR[SR] - VAR[SRI])^{-1/2} (SR - SRI) \quad (8)$$

Además, los resultados de los modelos de redes neuronales analizados serán comparados con los de un modelo ingenuo o AR(1), el cual se presenta en la ecuación (9).

$$\Delta IPSA_t = \alpha_1 \cdot \Delta IPSA_{t-1} + \varepsilon_t \quad (9)$$

donde Δ indica primera diferencia, ε representa el término de error, y α_1 es el coeficiente que acompaña a la variable explicativa.

4. Simulación y análisis de los resultados

En el caso del MLP recursivo, considerando un conjunto extramuestral de 50 observaciones, la red acertó en la predicción del signo de 31 de ellas, generando así una capacidad predictiva de un 62%. Esta red, tanto en su versión recursiva como estándar obtuvo el menor rendimiento, debido a que la simpleza de su estructura no fue capaz de generalizar bien este tipo de problema.

La red Ward recursiva fue la que obtuvo el mejor rendimiento ya que obtuvo una capacidad predictiva de un 72%. El mejor rendimiento con respecto al MLP recursivo se debe a que, en la capa oculta, la red Ward posee dos *Slabs* con distintas funciones de activación, lo que le permite a la capa de salida tener una visión del problema desde dos puntos de vista. Cabe destacar que, al momento de definir la arquitectura a utilizar para el problema, el software sugería un tipo de red Ward. Además, llama la atención que la mayor precisión se obtuviera con tasas de aprendizaje relativamente bajas para las primeras conexiones ya que, de acuerdo a la teoría, éstas debieran tener un alto valor al comienzo para ir disminuyendo en las conexiones posteriores.

La red de Jordan y Elman recursiva obtuvo una capacidad predictiva de un 64%. De acuerdo a la teoría, esta red era la más indicada para trabajar con series de tiempo debido a su capacidad para aprender secuencias. No obstante, la red Ward, a pesar de no ser una red destinada a resolver problemas de series de tiempo, obtuvo un mejor resultado.

Por su parte, el modelo ingenuo o AR(1) arrojó una capacidad predictiva del signo de las variaciones del IPSA de un 60%, superando levemente el desempeño mostrado por el MLP estándar. Los resultados se resumen en el cuadro 3.

Cuadro 3
Resultados de los modelos.

MODELO	CAPACIDAD PREDICTIVA	VALOR FINAL DE LA CARTERA	RENTABILIDAD ACUMULADA
Red Ward Recursiva	72%	124,42	24,42%
Red Jordan y Elman Estándar	66%	118,44	18,44%
Red Ward Estándar	66%	111,99	11,99%
Red Jordan y Elman Recursivo	64%	121,33	21,33%
Perceptrón Recursivo	62%	107,64	7,64%
Modelo Ingenuo (AR[1])	60%	107,06	18,31%
Perceptrón Estándar	58%	101,39	1,39%

A cada uno de estos modelos se les aplicó el test DA de Pesaran y Timmermann (1992), quienes mostraron que, bajo la hipótesis nula de que

$\hat{y}_{n+i}$ y y_{n+i} están independientemente distribuidos, dicho test sigue una distribución normal estándar. Los resultados del test se presentan en el cuadro 4.

Cuadro 4
Resultados del Test DA.

MODELO	ESTADÍSTICO DA	H ₀
Modelo Ingenuo	1.318	No se rechaza
Red Ward Estándar	2.220	Se rechaza
Red Ward Recursiva	3.117	Se rechaza
Perceptrón Estándar	1.104	No se rechaza
Perceptrón Recursivo	1.720	No se rechaza
Red Jordan y Elman Estándar	2.293	Se rechaza
Red Jordan y Elman Recursivo	1.985	Se rechaza

Los resultados muestran que la hipótesis nula no se rechaza en los casos del modelo ingenuo, Perceptrón estándar y Perceptrón recursivo, por lo que se concluye que éstos no pueden predecir el cambio direccional de la rentabilidad extramuestral del IPSA.

Lo anterior ha mostrado que el signo del retorno accionario es predecible en algún grado. No obstante, es necesario analizar si dicha capacidad predictiva se traduce en beneficios económicos significativos, por lo cual se calculó la rentabilidad acumulada que se hubiera obtenido si se hubiese comprado o vendido el IPSA siguiendo las recomendaciones del modelo de predicción, asumiendo una inversión inicial de US\$100⁶. Los resultados muestran que la mayor capacidad predictiva de la red Ward recursiva se materializa en una rentabilidad acumulada de un 24,42% (considerando 50 períodos semanales), la más alta de entre los modelos analizados. Por su parte, el MLP estándar, cuya predicción del signo fue la más baja (un 58%), generó la menor rentabilidad acumulada (un 1,39%).

⁶La inversión inicial es de US\$100. Por lo tanto, no se pueden comprar más unidades del índice que las que permite la inversión inicial, aún cuando el modelo de proyección siga recomendando comprar. De igual manera, sólo puede venderse aquello que ha sido comprado.

Cabe destacar que la red Jordan y Elman recursiva obtuvo una rentabilidad acumulada superior a la red Jordan y Elman estándar, 21,33% y 18,44% respectivamente, aún cuando ésta última poseía una mayor capacidad de predicción. A su vez, el modelo ingenuo generó una rentabilidad acumulada de 18,31%, mayor que la red Ward estándar y el MLP recursivo, a pesar de no evidenciar capacidad de predicción. Lo anterior permitiría concluir que no siempre existe una relación directa entre mayor capacidad predictiva y mayores retornos, lo que hace necesario evaluar la rentabilidad ex-post del modelo antes de su eventual utilización. Sin embargo, los resultados proveen evidencia que soporta la idea de que, para el caso del IPSA chileno, la red Ward recursiva y la red Jordan y Elman en su versión recursiva y estándar evidencian capacidad predictiva del cambio direccional del índice y producen retornos significativos en términos económicos, con respecto a los que generaría un modelo ingenuo. Cabe señalar que, al momento de calcular la rentabilidad de la cartera, los costos de transacción no fueron considerados⁷.

5. Conclusiones

De las diversas arquitecturas de redes neuronales elaboradas en este estudio, la red Ward recursiva fue la que obtuvo el mejor rendimiento en la predicción del signo de las variaciones del IPSA, de acuerdo al método de evaluación de la prueba del signo, alcanzando una capacidad predictiva extramuestral del 72%. A continuación se ubicó la red recurrente de Jordan y Elman estándar, con una capacidad predictiva de un 66%. Finalmente, se concluyó que el modelo ingenuo, el Perceptrón estándar y el Perceptrón recursivo no evidencian capacidad para predecir el cambio direccional de la rentabilidad extramuestral del IPSA.

Lo anterior ha mostrado que el signo del retorno accionario es predecible en algún grado. No obstante, es necesario analizar si dicha capacidad predictiva se traduce en beneficios económicos significativos.

⁷Si un indicador de análisis técnico desempeña su función, éste debe producir un retorno que exceda la suma de los costos de transacción más la rentabilidad negativa originada por sus señales falsas.

Los resultados muestran que la mayor capacidad predictiva de la red Ward recursiva se materializa en una rentabilidad acumulada de un 24,42%, la más alta de entre los modelos analizados. Por su parte, el MLP estándar, cuya predicción del signo fue la más baja (un 58%), generó la menor rentabilidad acumulada (un 1,39%). A su vez, el modelo ingenuo generó una rentabilidad acumulada de 18,31%, mayor que la red Ward estándar y el MLP recursivo, a pesar de no evidenciar capacidad de predicción. Lo anterior permitiría concluir que no siempre existe una relación directa entre mayor capacidad predictiva y mayores retornos, lo que hace necesario evaluar la rentabilidad ex-post del modelo antes de su eventual utilización. Sin embargo, los resultados proveen evidencia que soporta la idea de que, para el caso del IPSA chileno, la red Ward recursiva y la red Jordan y Elman en su versión recursiva y estándar pueden predecir el cambio direccional del índice y producen retornos significativos en términos económicos, con respecto a los que generaría un modelo ingenuo. Cabe señalar que, al momento de calcular la rentabilidad de la cartera, los costos de transacción no fueron considerados.

De esta forma se mostró que la capacidad de los modelos de redes neuronales resulta relevante para predecir el signo de las variaciones del IPSA chileno, lo que los sitúa como una alternativa al análisis técnico y a los modelos ingenuos aplicados sobre series de tiempo, al momento de predecir la evolución del mercado para tomar decisiones de inversión o desinversión en activos bursátiles. Más aún cuando, de acuerdo al estudio de Leung, Daouk y Chen (2000), la predicción de la dirección del movimiento del índice del mercado accionario es útil para desarrollar estrategias de transacción efectivas, las cuales pueden arrojar mejores resultados que aquellas basadas en la proyección del valor de la variable observada. Así, el empleo de redes neuronales podría orientar la conformación de portafolios de inversión y la forma en que éstos podrían ser estructurados, a fin de aprovechar las alzas pronosticadas para el mercado y cubrirse ante las bajas de éste, dadas las expectativas de eventos futuros y la relación riesgo-retorno que se espera obtener.

ANEXO 1. Descripción de la metodología de estimación

A continuación se presenta el proceso mediante el cual el Perceptrón Multicapa estándar obtiene la salida (predicción) y actualiza los pesos de conexión entre las capas. De acuerdo a Freeman y Skapura (1993), la aplicación del algoritmo de aprendizaje de retropropagación del error se realiza de la siguiente forma:

- Se aplica el vector de datos $x_p = [x_{p1}, x_{p2}, \dots, x_{pN}]$ a las unidades de entrada, el cual contiene las variables que se emplearán para predecir el signo de las variaciones del IPSA. x_p corresponde al vector de datos de entrada relativos al período "p", x_{pi} representa el valor de la "i-ésima" variable explicativa en el período "p" de la serie de tiempo, y el superíndice "t" indica traspuesta.
- Cada p-ésimo vector se emplea para calcular los valores de entrada neta para las unidades de la capa oculta. Así, cada neurona de la capa oculta calcula el valor de entrada neto como la sumatoria de los valores de entrada ponderados por sus respectivos pesos, como se indica en la ecuación (1):

$$Neta_{pj}^h = \sum_{i=1}^N w_{ji}^h \cdot x_{pi} + \theta_j^h \quad (1)$$

donde $Neta_{pj}^h$ es el valor de entrada neto para la j-ésima neurona de la capa oculta⁸ relacionada al período "p", w_{ji}^h es la ponderación que recibe la i-ésima variable explicativa asociada a la j-ésima neurona de

⁸Un punto importante en el diseño de una red neuronal consiste en determinar el número apropiado de neuronas en la capa oculta, el cual está relacionado con la capacidad de mapeo de la red. Mientras más grande es su número, mayor es la capacidad del modelo para memorizar el conjunto de entrenamiento. Sin embargo, si se continúa aumentando el tamaño de la red, hay un punto en donde la generalización (la capacidad de proporcionar una respuesta correcta ante datos no utilizados en su entrenamiento) empeorará, debido a que puede ocurrir un sobreajuste al conjunto de entrenamiento. Desafortunadamente, no hay una respuesta consistente a esta cuestión, y la forma más eficiente para determinar el tamaño óptimo de la red es a través de la experimentación. No obstante, Salchenberger, Cinar y Lash (1992) recomiendan que el número de neuronas de la capa oculta debe corresponder a aproximadamente el 75% de las unidades de entrada.

la capa oculta, la cual puede ser excitatoria (pesos positivos) o inhibitoria (pesos negativos), x_{ij} corresponde al valor de la i -ésima variable de entrada asociada a la j -ésima neurona de la capa oculta, y θ_j^h es el bias (sesgo), el cual representa los grados de libertad y se comporta como otro ponderador. El superíndice "h" acompaña a los valores relacionados con la capa oculta.

De esta forma se tendrá una matriz de pesos, como se muestra en la expresión (2):

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{1n} \\ w_{21} & w_{22} & w_{23} & \cdots & w_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ w_{m1} & w_{m2} & w_{m3} & \cdots & w_{mn} \end{bmatrix}^h, \text{ con } j=1, \dots, n; i=1, \dots, m \quad (2)$$

$(i) \times (j)$

- c. Una vez que la sumatoria ha sido calculada, ésta debe ser procesada por una función de activación (lineal, tangente, logística o gaussiana). En consecuencia, se aplica una función de activación sobre el valor de entrada neto de cada una de las neuronas que conforman la capa oculta. Así se obtiene la salida de la capa oculta, i_{pj}^h , como se muestra en la ecuación (3):

$$i_{pj}^h = f_j^h(\text{Neta}_{pj}^h) \quad (3)$$

i_{pj}^h representa el valor de salida de la j -ésima neurona perteneciente a la capa oculta y f_j^h es la función de activación aplicada sobre el valor de entrada neto de la j -ésima neurona de la mencionada capa.

- d. Luego, i_{pj}^h pasa a ser la entrada de la capa de salida. Nuevamente se calculan los valores netos de las entradas para cada unidad o neurona de la capa de salida, como lo señala la ecuación (4):

$$\text{Neta}_{pk}^o = \sum_{j=1}^L w_{kj}^o \cdot i_{pj}^h + \theta_k^o \quad (4)$$

donde $Neta_{pk}^o$ es el valor de entrada neto (relacionado al período "p") para la k-ésima neurona de la capa de salida, w_{kj}^o es la ponderación que recibe el j-ésimo valor de salida de la capa oculta asociado a la k-ésima neurona de la capa de salida. θ_k^o es el bias, el cual representa los grados de libertad y se comporta como otro ponderador, y el superíndice "o" acompaña a los valores relacionados con la capa de salida. De esta forma se tendrá una matriz de pesos, como se muestra en la expresión (5):

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{1n} \\ w_{21} & w_{22} & w_{23} & \cdots & w_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ w_{l1} & w_{l2} & w_{l3} & \cdots & w_{ln} \end{bmatrix}^o, \text{ con } j=1, \dots, n; k=1, \dots, l \quad (5)$$

(k) x (j)

- e. Posteriormente se obtienen las salidas de la capa oculta, o_{pk} en la ecuación (6), aplicándose la función de activación sobre el valor neto previo, $Neta_{pk}^o$. f_k^o es la función de activación aplicada sobre el valor de entrada neto de la k-ésima neurona de la capa de salida:

$$o_{pk} = f_k^o(Neta_{pk}^o) \quad (6)$$

La función de activación en esta fase no necesariamente será la misma que la aplicada en la capa oculta.

- f. Se calculan los términos de error para las unidades de salida, δ_{pk}^o , como se muestra en la ecuación (7). y_{pk} representa los valores observados de la variable de salida en el momento "p", o_{pk} representa el valor de la predicción para el período "p" generado por la k-ésima neurona de la capa de salida, y $f_k^{o'}(neta_{pk}^o)$ corresponde a la primera derivada de la función de activación aplicada sobre la capa de salida:

$$\delta_{pk}^o = (y_{pk} - o_{pk}) f_k^{o'}(neta_{pk}^o) \quad (7)$$

- g. Se calculan los términos de error para las neuronas de la capa oculta: el error total se distribuye entre las neuronas de la capa oculta en proporción al aporte que cada una de ellas realizó al error total, como se señala en la ecuación (8):

$$\delta_{pj}^h = f_j^h (Neta_{pj}^h) \sum_k \delta_{pk}^o \cdot w_{kj}^o \quad (8)$$

El término δ_{pj}^h representa la fracción del error total de la predicción realizada para el período "p", asociado a la j-ésima neurona de la capa oculta.

- h. Una vez que se ha distribuido el error entre las neuronas de la capa oculta, se actualizan los pesos que recibe el j-ésimo valor de salida de la capa oculta asociado a la k-ésima neurona de la capa de salida, $w_{kj}^o(t)$. Esta actualización se hace en función de la tasa de aprendizaje, η , del error estimado para cada neurona de la capa de salida, δ_{pk}^o , y del valor de salida de la j-ésima neurona perteneciente a la capa oculta, i_{pj}^h . El peso actualizado, $w_{kj}^o(t+1)$ se expresa a través de la ecuación (9):

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \eta \cdot \delta_{pk}^o \cdot i_{pj}^h \quad (9)$$

- i. Se actualizan los pesos asociados a la j-ésima neurona de la capa oculta, w_{ji}^h , en función de la tasa de aprendizaje, η , del error obtenido para cada neurona de la capa oculta, δ_{pj}^h , y del valor de la i-ésima variable de entrada, x_i , como se indica en la ecuación (10):

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \eta \cdot \delta_{pj}^h \cdot x_i \quad (10)$$

- j. Finalmente, se calcula el termino de error a través de la expresión (11):

$$E_p = \frac{1}{2} \cdot \sum_{k=1}^M (\delta_{pk}^o)^2 \quad (11)$$

- k. El proceso descrito a partir del paso (a) se repite para un nuevo vector de datos de entrada, $x_p = [x_{p1}, x_{p2}, \dots, x_{pN}]$, y finaliza con el cálculo del error de predicción explicado en el paso (j). Cuando el proceso se haya realizado para cada uno de los vectores⁹ que componen el conjunto de datos de entrenamiento, se habrá concluido una época. A su vez, pueden completarse tantas épocas como se considere necesario durante la experimentación.

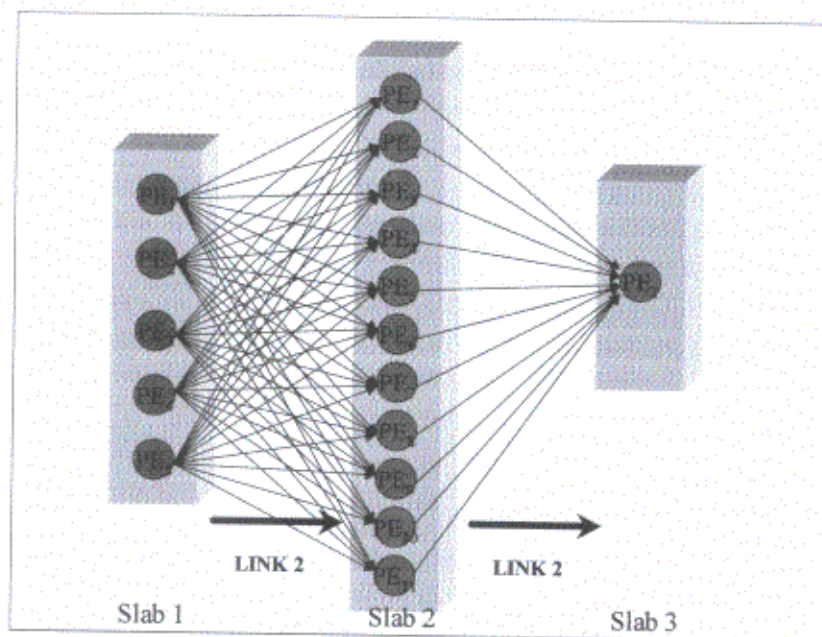
⁹El número de vectores será igual al número de datos que contengan las series de tiempo que forman parte del conjunto de entrenamiento.

Anexo 2

Arquitectura de la red neuronal

En el caso del MLP estándar, luego de probar con numerosas arquitecturas, funciones de activación, número de neuronas en la capa oculta, reglas de aprendizaje y sus valores, la mejor combinación de estos parámetros se obtuvo con una red de tres capas (capa de entrada, capa oculta y capa de salida), con 1 *Slab* en cada una de ellas (ver figura 2.1).

Figura 2.1
MLP estándar



La capa de entrada (*Slab 1*) consta de 5 neuronas, número que se determinó de acuerdo al número de variables de entradas. En la capa oculta (*Slab 2*), el mejor rendimiento se obtuvo con 11 PEs, ya que un número mayor sólo disminuía el error en el conjunto de entrenamiento y no en el conjunto extramuestral. La salida neta de la capa oculta viene dada por la ecuación (1):

$$i_{pj}^h = f_j^h(Neta_{pj}^h) \quad , \text{ con } j=1, 2, \dots, 11. \quad (1)$$

donde:

$$Neta_{pj}^h = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \dots & w_{15} \\ w_{21} & w_{22} & w_{23} & \dots & w_{25} \\ w_{31} & w_{32} & w_{33} & \dots & w_{35} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ w_{111} & w_{112} & w_{113} & \dots & w_{115} \end{bmatrix}_{(11 \times 5)}^h \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_5 \end{bmatrix}_{(5 \times 1)} + \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_{11} \end{bmatrix}_{(11 \times 1)}^h = \begin{bmatrix} Neta_1 \\ Neta_2 \\ Neta_3 \\ \vdots \\ Neta_{11} \end{bmatrix}_{(11 \times 1)}^h \quad (2)$$

La capa de salida (*Slab 3*) consta de un sólo PE, el cual se determinó de acuerdo al número de variables de salida. En consecuencia, la salida neta de la capa de salida queda expresada de acuerdo a la ecuación (3):

$$o_{pk} = f_k^o(Neta_{pk}^o) \quad (3)$$

donde:

$$Neta_{pk}^o = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \dots & w_{111} \end{bmatrix}_{(1 \times 11)}^o \times \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ \vdots \\ i_{11} \end{bmatrix}_{(11 \times 1)}^h + \begin{bmatrix} \theta_1 \end{bmatrix}_{(1 \times 1)}^o = \begin{bmatrix} Neta_1 \end{bmatrix}_{(1 \times 1)}^o \quad (4)$$

El *Link 1* y el *Link 2* emplean la regla de aprendizaje de propagación hacia atrás. Los cuadros 2.1 y 2.2 muestran las características de los *Slabs* y los parámetros de la red, respectivamente. Estos últimos son menores en la conexión dos que en la conexión uno, ya que el error tiende a hacerse más grande hacia la capa de salida y es necesario que las iteraciones sean menores. Los pesos iniciales para cada conexión son de 0,3.

Cuadro 2.1

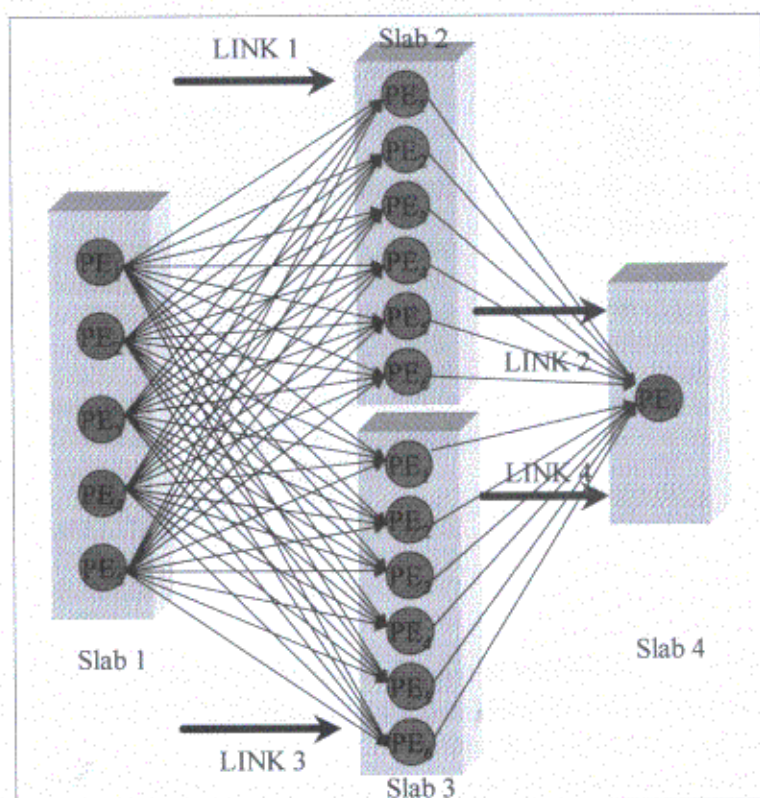
	MLP ESTÁNDAR				RED WARD				RED JORDAN-ELMAN			
	SLAB 1	SLAB 2	SLAB 3	SLAB 4	SLAB 1	SLAB 2	SLAB 3	SLAB 4	SLAB 1	SLAB 2	SLAB 3	SLAB 4
Nº Pes	5	11	1	5	5	6	6	1	5	11	11	1
F. Activación	Tangente	Tangente	Logistic	Tangente	Tangente	Gaussiana	Gaussiana	Logistic	Tangente	Tangente	Logistic	Complemento

Cuadro 2.2

	MLP ESTÁNDAR				RED WARD				RED JORDAN-ELMAN			
	LINK 1	LINK 2	LINK 1	LINK 2	LINK 1	LINK 2	LINK 3	LINK 4	LINK 1	LINK 2	LINK 3	LINK 4
Tasa aprendizaje	0.5	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.1	0.01	0.1	0.1
Tasa momento	0.7	0.1	0.01	0.01	0.01	0.01	0.01	0.01	0.1	0.1	0.1	0.1

En el caso de la red Ward, nuevamente se probaron diferentes combinaciones de parámetros y arquitecturas, siendo la más adecuada una red de tres capas (de entrada, oculta y de salida). Las capas de entrada y salida poseen un *Slab*, mientras que la capa oculta cuenta con dos *Slabs* (ver figura 2.2).

Figura 2.2
Red Ward



En el cuadro 2.1 se muestra que la capa de entrada (*Slab 1*) posee 5 PEs, uno por cada variable de entrada, la capa oculta posee 12 PEs, los cuales se dividen en partes iguales entre los *Slab 2* y *3*, y la capa de salida (*Slab 4*)

posee un solo PE. De esta forma, la salida neta de la capa oculta viene dada por la ecuación (5):

$$i_{pj}^h = f_j^h (Neta_{pj}^h), \quad \text{con } j=1, 2, \dots, 12. \quad (5)$$

donde la expresión (6) se emplea para los Slabs 2 y 3, respectivamente:

$$Neta_{pj}^h = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \dots & w_{15} \\ w_{21} & w_{22} & w_{23} & \dots & w_{25} \\ w_{31} & w_{32} & w_{33} & \dots & w_{33} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ w_{61} & w_{62} & w_{63} & \dots & w_{65} \end{bmatrix}_{(6 \times 5)}^h \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_5 \end{bmatrix}_{(5 \times 1)} + \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_6 \end{bmatrix}_{(6 \times 1)}^h = \begin{bmatrix} Neta_1 \\ Neta_2 \\ Neta_3 \\ \vdots \\ Neta_6 \end{bmatrix}_{(6 \times 1)}^h \quad (6)$$

La salida neta de la capa de salida queda expresada de acuerdo a la ecuación (7), para los Slabs 2 y 3:

$$o_{pk} = f_k^o (Neta_{pk}^o) \quad (7)$$

donde:

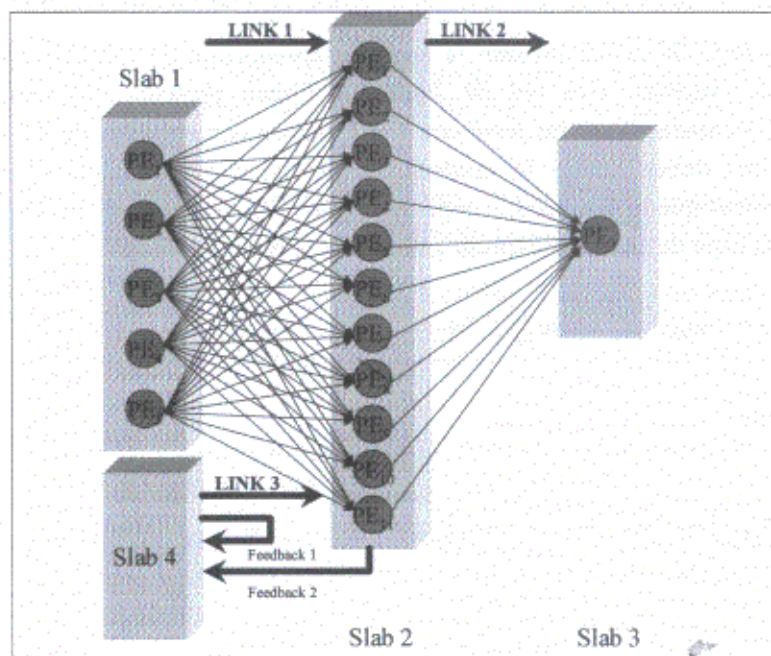
$$Neta_{pk}^o = [w_{11} \quad w_{12} \quad w_{13} \quad \dots \quad w_{16}]_{(1 \times 6)}^o \times \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ \vdots \\ i_6 \end{bmatrix}_{(6 \times 1)}^h + [\theta_1]_{(1 \times 1)}^o = [Neta_1]_{(1 \times 1)}^o \quad (8)$$

Los Links 1, 2, 3 y 4 utilizan la regla de aprendizaje de propagación hacia atrás y sus parámetros se presentan en el cuadro 2.2. Esta red también cuenta con pesos iniciales de 0,3.

Finalmente, entre las redes con conexiones recurrentes (Jordan y Elman), la que obtuvo el mejor rendimiento fue la realimentada en la capa oculta, la cual cuenta con tres capas (de entrada, oculta y de salida). Esta red se caracteriza por poseer un Slab extra en la capa de entrada, la cual contiene las características detectadas en los datos de patrones anteriores,

permitiendo el empleo de rezagos de las variables de entrada (ver figura 2.3).

Figura 2.3
Red de Jordan y Elman



El cuadro 1 contiene las características de las capas de la red. Cabe señalar que el *Slab 4*, que se puede considerar como parte de la capa de entrada, posee 11 PEs y no presenta función de activación, ya que está estrechamente ligada al *Slab 2*, el cual contiene el mismo número de PEs. Luego, la salida neta de la capa oculta viene dada por la ecuación (9):

$$i_{pj}^h = f_j^h (Neta_{pj}^h), \quad \text{con } j=1, 2, \dots, 11. \quad (9)$$

donde:

$$Neta_{pj}^h = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{15} \\ w_{21} & w_{22} & w_{23} & \cdots & w_{25} \\ w_{31} & w_{32} & w_{33} & \cdots & w_{35} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ w_{111} & w_{112} & w_{113} & \cdots & w_{115} \end{bmatrix}_{(11 \times 5)}^h \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_5 \end{bmatrix}_{(5 \times 1)} + \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta_{11} \end{bmatrix}_{(11 \times 1)}^h = \begin{bmatrix} Neta_1 \\ Neta_2 \\ Neta_3 \\ \vdots \\ Neta_{11} \end{bmatrix}_{(11 \times 1)}^h \quad (10)$$

Los *links* de la red utilizan la regla de aprendizaje de propagación hacia atrás. Además, esta red de Jordan y Elman presenta dos conexiones realimentadas llamadas *feedback* 1 y 2, ambas con un valor de 0,5 (ambos valores deben sumar la unidad). El *feedback* 1 realimenta el *Slab* 4, mientras que el *feedback* 2 conecta el *Slab* 2 con el *Slab* 4. La salida neta de la capa de salida se expresa a través de la ecuación (11):

$$o_{pk} = f_k^o(Neta_{pk}^o) \quad (11)$$

donde:

$$Neta_{pk}^o = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{111} \end{bmatrix}_{(1 \times 11)}^p \times \begin{bmatrix} i_1 \\ i_2 \\ i_3 \\ \vdots \\ i_{11} \end{bmatrix}_{(11 \times 1)}^h + \begin{bmatrix} \theta_1 \end{bmatrix}_{(1 \times 1)}^p = \begin{bmatrix} Neta_1 \end{bmatrix}_{(1 \times 1)}^p \quad (12).$$

REFERENCIAS

- BOSARGE, W.E. (1993). "Adaptive processes to exploit the nonlinear structure of financial markets". In R.R. Trippi & E. Turban (eds.), *Neural networks in finance and investing*. New York, Irwin, pp. 371-402.
- BROCKETT, P. W., COOPER W. W., GOLDEN L. L. y PITAKTONG U. (1994). "A neural network method for obtaining an early warning of insurer insolvency". *The Journal of Risk and Insurance* 6, pp. 402-424.
- CHEN, A. y LEUNG, M. (1998). "Dynamic foreign currency trading guided by adaptive forecasting". *Review of Pacific Basin Financial Markets and Policies* 1, pp. 383-418.
- CHOPRA, N., LAKONISHOK J. y RITTER J. R. (1992). "Measuring abnormal returns: Do stocks overreact?" *Journal of Financial Economics*, 31, 235-268.
- DASGUPTA, C. G., DISPENSA, G. S. y GHOSE, S. (1994). "Comparing the predictive performance of a neural network model with some traditional market response models". *International Journal of Forecasting* (10)2, pp. 235-244.
- DEBONDT, W. F. M. y THALER R. (1985). "Does the stock market overreact?" *The Journal of Finance* 40, 793-805.
- ESTRELLA, A., y MISHKIN, F. S. (1998). "Predicting US recessions: Financial variables as leading indicators". *The Review of Economics and Statistics* 80(1), pp. 45-61.
- FAMA, E. (1970). "Efficient capital markets: A review of theory and empirical work". *The Journal of Finance* 25, p.p. 383-417.
- FAMA, E. y FRENCH K. R. (1988). "Permanent and temporary components of stock prices". *Journal of Political Economy* 98, 247-273.
- FERSON, W. y HARVEY, C. (1991). "The variation of economic risk premiums". *Journal of Political Economy* 99, pp. 385-415.
- FREEMAN, J. y SKAPURA, D. (1993). "Redes neuronales: Algoritmos, aplicaciones y técnicas de programación". Primera Edición. Copublicación de Addison-Wesley Iberoamericana, S.A. y Ediciones Díaz de Santos, S.A. Wilmington, Delaware, USA. 431 pp.

- GORR WILPEN L. (1994). "Research prospective on neural network forecasting". *International Journal of Forecasting* (10)1, pp. 1-4.
- HAWLEY, D., JOHNSON, J., y RAINA, D. (1990). "Artificial neural systems: A new tool for financial decision-making". *Financial Analysts Journal* 23, pp. 63-72.
- HERBRICH, R., KEILBACH, M., GRAEPEL, T., BOLLMANN-SDORRA, P. y OBERMAYER K. (2000). "Neural networks in economics: Background, applications and new developments". In T. Brenner, editor, *Advances in Computational Economics: Computational Techniques for Modeling Learning in Economics*, volume 11, pages 169-196. Kluwer Academics.
- HILL, T., MARQUEZ, L., O'CONNOR, M. y REMUS, W. (1994). "Artificial neural network models for forecasting and decision making". *International Journal of Forecasting* (10)1, pp. 5-15.
- HODGSON, A. y NICHOLLS, D. (1991). "The impact of index futures markets on Australian share market volatility". *Journal of Business Finance and Accounting* 18, pp. 267-280.
- HORNIK, K., STINCHCOMBE M. y WHITE H. (1989). "Multilayer feedforward networks are universal approximators". *Neural Networks*, 2(5), pp. 359-366.
- JAGIELSKA, I. y JAWORSKI, J. (1996). "Neural network for predicting the performance of credit card accounts". *Computational Economics*, 9, pp. 77-82.
- JEGADEESH, NARASIMHAN (1990). "Evidence of predictable behavior of security returns". *The Journal of Finance*, Vol. 45(3), pp. 881-898 (July).
- KANAS, ANGELOS (2001). "Neural networks linear forecasts for stocks returns". *International Journal of Finance and Economics*, 6, pp. 245-254.
- KUO, C. y REITSCH A. (Winter 1995-96). "Neural networks vs. conventional methods of forecasting", *Journal of Business Forecasting* 14, N°4, pp.17-22.
- LEUNG MARK T., DAOUK HAZEM y CHEN AN-SING (2000). "Forecasting stock indices: a comparison of classification and level estimation models". *International Journal of Forecasting* (16)2, pp.173-190.

- LO, A. y MAC KINLEY A. C. (1988). "Stock market price do not follow random walk: Evidence from a simple specification test". *Review of Financial Studies* 1, pp. 41-66.
- MABERLY, E. D. (1986). The informational content of the interday price change with respect to stock index futures. *Journal of Futures Markets* 6, pp. 385-395.
- MCCLELLAND, J. y RUMELHART, D. (1986). "Explorations in parallel distributed processing", vol. 1 and 2. MIT Press, Cambridge, MA.
- MAROSE, R. A. (1990). "A financial neural network application". *AI Expert*, May, pp. 50-53.
- MARTÍN DEL BRÍO, B. & SANZ, A. (1997). "Redes neuronales y sistemas borrosos: Introducción, teórica y práctica". Primera Edición. Ra-ma. 387 pp.
- O'CONNOR, M., REMUS, W., y GRIGGS, K. (1997). "Going up-going down: How good are people at forecasting trends and changes in trends?" *Journal of Forecasting* 16, pp. 165-176.
- ODOM M. D. y SHARDA R. (1990). "A neural network model for bankruptcy prediction". Proceedings of the IEEE International Conference on Neural Networks, San Diego II, pp. 163-168.
- PARKER, D. B. (1985). "Learning logic. Technical Report TR-47". Center for Computational Research in Economics and Management Science, MIT, Cambridge, MA (April).
- PESARAN, M.H. & TIMMERMANN A. (1992). "A simple nonparametric test of predictive performance". *Journal of Business and Economic Statistics* 10, pp. 461-465.
- POTERBA, J. M. y Summers L. H. (1988). "Mean reversion in stock prices: Evidence and implications". *Journal of Financial Economics* 22, pp. 27-59.
- QI, MIN (2001). "Predicting US recessions with leading indicators via neural network models". *International Journal of Forecasting*, (17)3, pp. 383-401.

- RAHIMIAN E., SINGH S., THAMMACHOTE T. y VIRMANI R. (1993). "Bankruptcy prediction by neural network". In: R. R. Trippi & E. Turban (eds.): *Neural networks in finance and investing*. Probus Publishing, pp. 159-171.
- REFENES, A. P. (1995). "Neural networks in the capital markets", Wiley, New York.
- RUMELHART, D., HINTON G. y WILLIAMS J. (1986). "Learning representations by back-propagating errors". *Nature*, 323 (Oct 9), pp. 533-536.
- SHARDA, R. (1994). "Neural networks for the MS/OR analysis: An application bibliography". *Interfaces*, 24, pp. 116-130.
- TANG, Z., DE ALMIEDA C. y FISHWICK P. (1991). "Time-series forecasting using neural networks vs. Box-Jenkins methodology". *Simulation*, 57(5), pp. 303-310.
- TSIBOURIS, G. y ZEIDENBERG, M. (1995). "Testing the efficient markets hypothesis with gradient descent algorithms". In A.P. Refenes (ed.), *Neural networks in the capital markets*. Chichester, UK, Wiley, pp. 127-136.
- WERBOS, P. (1974). "Beyond regression: New tools for prediction and analysis in the behavioral sciences". Tesis doctoral, Harvard, Cambridge, MA, (Agosto).
- WHITE, H. (1993). "Economic prediction using neural networks: the case of IBM daily stock returns". In R.R. Trippi & E. Turban (eds.), *Neural networks in finance and investing*. New York, Irwin, pp. 315-328.
- WIDROW, B., RUMELHART, D.E. y LEHR, M.A. (1994). "Neural networks: Applications in industry, business and science". *Communications of the ACM*, 37, 3 (March), pp.93-105.
- WILSON, J. H. y KEATING B. (1998). *Business Forecasting*. Third Edition, U.Irwin/McGraw-Hill, U.S.A.
- WILSON, R. L. y SHARDA R. (1994). "Bankruptcy prediction using neural networks". *Decision Support Systems*, 11(5), June, pp. 545-557.
- WU, Y., y ZHANG, H. (1997). "Forward premiums as unbiased predictors of future currency depreciation: A non-parametric analysis". *Journal of International Money and Finance* 16, pp. 609-623.

- YANG, Z. (1999). "Probabilistic neural networks in bankruptcy prediction". *Journal of Business Research* 44, p.p. 67-74.
- YODA, M. (1994). "Predicting the Tokyo stock market". In G.J. Deboeck (ed.), *Trading on the edge: Neural, genetic, and the fuzzy systems for chaotic financial markets*. New York, Wiley, pp. 66-79.