

**“REDES NEURONALES APLICADAS A LA
PREDICIÓN DEL PRECIO DEL ORO Y
MEDICIÓN DE LA ROBUSTEZ DE LOS
RESULTADOS UTILIZANDO BOOTSTRAP”**

Seminario Para Optar al Título de Ingeniero Comercial Mención Administración

RODOLFO FRANCISCO FRIZ ECHEVERRÍA.

Profesor Guía: Antonino Parisi F. Ph.D.

Santiago, noviembre de 2003

..	1
I. INTRODUCCIÓN . .	3
II. OBJETIVOS . .	5
A. OBJETIVOS GENERALES . .	5
B. OBJETIVOS ESPECIFICOS . .	5
III. MARCO TEORICO .	7
A. REDES NEURONALES .	7
1. Las Neuronas . .	10
2. Funciones de Activación .	11
3. Características de la Redes Neuronales Artificiales (R.N.A.) .	15
4. Clasificación de las Redes Neuronales . .	16
5. Tipos de funcionamiento de las Redes Neuronales Artificiales .	17
6. Tipos de Redes Neuronales Artificiales .	21
7. Ventajas de las Redes Neuronales Artificiales . .	32
8. Desventajas de las Redes Neuronales .	33
IV. MODELOS DE PREDICCIÓN DE REDES NEURONLAES . .	35
PARTE I: MODELOS ROLLING V/S RECURSIVOS .	37
A. METODOLOGIA . .	37
1. Variables de entrada .	37
2. Modelo .	38
B. Red Neuronal con funcionamiento recursivo y rolling .	39
1. Funcionamiento Recursivo . .	42
2. Funcionamiento Rolling .	42
C. Evaluación de la capacidad predictiva .	43
1. Análisis De Resultados . .	43
2. Conclusiones parte I .	44
PARTE II. MEDICIÓN DE LA ROVUSTED UTILIZANDO BOOTSTRAP . .	45

A. METODOLOGÍA . .	45
1. Modelo .	45
2. Bootstrap .	46
B. ANÁLISIS DE LOS RESULTADOS .	46
1. Conclusiones Parte II .	50
V. CONCLUSIONES FINALES .	51
VI. BIBLIOGRAFIA .	53
Anexos. .	59

A mi familia, en especial a mi madre por todo su apoyo. A Ximena Esteban, por todo su cariño y comprensión. A todo el equipo Parisinet, gracias por su amistad.

I. INTRODUCCIÓN

El oro ha sido considerado como un activo refugio, por la cualidad que tiene de ser tranzado en cualquier época de la historia, bajo cualquier contexto político y las circunstancias más diversas.

En el año 1971, tras la intervención del presidente Nixon como forma de paliar las presiones inflacionarias, pierde vigencia el tratado de Bretton Woods. La liberación del patrón oro provocó una escalada de precios nunca antes vista, incentivando a los productores de petróleo a aumentar el precio del barril de crudo, lo cual derivó finalmente en la crisis del petróleo. Este hecho histórico significó el inicio de una gran volatilidad en el precio del oro, especialmente explicada en épocas de inestabilidad, transformándolo en un indicador vital para los mercados financieros.

El oro probó ser la manera más efectiva de recaudar efectivo durante el crash de la bolsa de 1987, y una vez más en 1997 y 1998 durante la crisis de deuda asiática. En ambas guerras del golfo, la demanda por oro aumentó significativamente, por lo que mantener una pequeña proporción del portfolio en oro puede ser invaluable en momentos cuando el efectivo es esencial, tanto como para cubrir *margin calls* u otras necesidades.

Dado lo anterior es muy importante obtener una capacidad predicativa sobre este activo, para poder protegerse de mejor manera de los ciclos económicos, podemos ver que se realizan importantes esfuerzos para poder obtener una cierta capacidad predicativa, aplicando diferentes técnicas predicativas, por ser el oro un indicador del comportamiento futuro de las economías. Esto responde a la necesidad de disminuir el riesgo y maximizar la utilidad futura.

Es así como se ha pasado de técnicas de modelamiento lineal, como la minimización el error cuadrático medio a técnicas no lineales para la predicción de los precios bursátiles, entre estas últimas podemos destacar a las Redes Neuronales, las cuales imitan el funcionamiento del cerebro humano, aprendiendo de los eventos anteriores, trabajando en forma paralela en la solución del problema, para luego obtener la mejor predicción posible, para tomar una decisión responsable e informada.

Dentro de este nuevo paradigma de las Redes Neuronales encontramos a las redes reconstruidas tipo Recursivo y tipo Rolling, con las cuales se ha logrado una mayor capacidad predictiva y un uso eficiente de toda la información disponible en el mercado.

Este estudio tiene por objeto determinar en una primera parte, la capacidad predictiva de las redes neuronales tanto Rolling como Recursivas en la predicción de signos de la variación del precio del oro, para en una segunda parte determinar si los resultados obtenidos por una de estas redes es robusto en los distintos escenarios económicos o sea si se obtiene la misma predicción de signos bajo diferentes escenarios ficticios, los cuales se simulan con la técnica Bootstrap, para obtener una distribución de los retornos de la técnica.

II. OBJETIVOS

A. OBJETIVOS GENERALES

1. Crear un modelo para predecir la variación del precio del oro y lograr una alta capacidad predictiva.
2. Realizar un estudio comparativo de la predicción con Redes Neuronales al utilizar las técnicas Recursivo o Rolling.
3. Medir la robustez de los resultados aplicando bootstrap.

B. OBJETIVOS ESPECIFICOS

1. Diseñar una base de datos con las distintas variables financieras que mejor explican las variaciones del precio del oro.
2. Buscar el tamaño muestral de los modelos de Redes Neuronales Dinámicas que obtenga la mayor capacidad predictiva.

3. Compara que funcionamiento, si Rolling o Recursivo, entrega mejores resultados en la predicción de la variación del precio del oro. 3.

Medir la significancia estadística de los resultados obtenidos, por los distintos modelos 4.

Medir la robustez de los resultados aplicando bootstrap, obteniendo distribuciones de los resultados, intervalos de confianzas y la seguridad de que el modelo se aplica no solo al periodo de tiempo estudiado sino que a cualquier escenario económico. 5.

III. MARCO TEORICO

A. REDES NEURONALES

Las Redes Neuronales son un nuevo paradigma en procesamiento de datos con grandes perspectivas para manejar muchos de los más desafiantes problemas computacionales. Mientras las técnicas de inteligencia artificial han ayudado a expandir las capacidades de los computadores existentes automatizando los procesos de razonamiento humano a nivel de *software*, las redes neuronales son un intento de imitar las capacidades de procesamiento del cerebro a nivel de *hardware*; esto es, a nivel de las neuronas individuales y su comportamiento general cuando se combinan en grandes redes altamente interconectadas.

Las redes neuronales se han aplicado en numerosas áreas que involucran el procesamiento de señales, el reconocimiento de patrones complejos y las tareas de clasificación, así como el procesamiento de imágenes, el reconocimiento de voz y caracteres, filtrado de ruido, compresión de datos, predicción de series de tiempo y optimización. Otra importante área de investigación es el control automático, donde las redes neuronales han sido investigadas para el control adaptativo de robots y vehículos autónomos, y la identificación y control de procesos complejos o desconocidos. La capacidad de las redes neuronales para el aprendizaje adaptativo, mapeo de señales no

lineales y tolerancia a fallas están siendo explotadas para llenar muchas de las persistentes fallas presentes en las técnicas clásicas de control.¹

Según Martín del Brío y Sanz una definición de redes neuronales sería la siguiente:

“Las redes neuronales artificiales son sistemas de procesamiento que copian esquemáticamente la estructura neuronal del cerebro para tratar de reproducir sus capacidades, como por ejemplo, aprender de la experiencia a partir de las señales o datos del exterior, con el objetivo de construir sistemas de procesamiento de la información paralelos, distribuidos y adaptativos”.²

Las redes neuronales son sistemas de procesamiento que tratan de reproducir la capacidad que tiene el cerebro humano para reconocer patrones, hacer predicciones o tomar una decisión en base a la experiencia pasada. Así, mientras el cerebro humano confía en la estimulación neuronal, las redes neuronales actúan sobre un conjunto de datos para aprender. Dichas redes destacan por su desempeño en la solución de complejos problemas, donde el reconocimiento de modelos o comportamientos es importante.

Uno de los primeros modelos matemáticos de una neurona fue el propuesto por McCulloch y Pitts en 1943. En este modelo, ilustrado en la figura a continuación, cada neurona consta de un conjunto de entradas, S_i , y una sola salida S_j . Cada entrada i está afectada por un coeficiente que se denomina peso y se representa por la letra W_{ij} . El subíndice i refleja que el peso afecta a la entrada i , y el subíndice j que se trata de la neurona j .

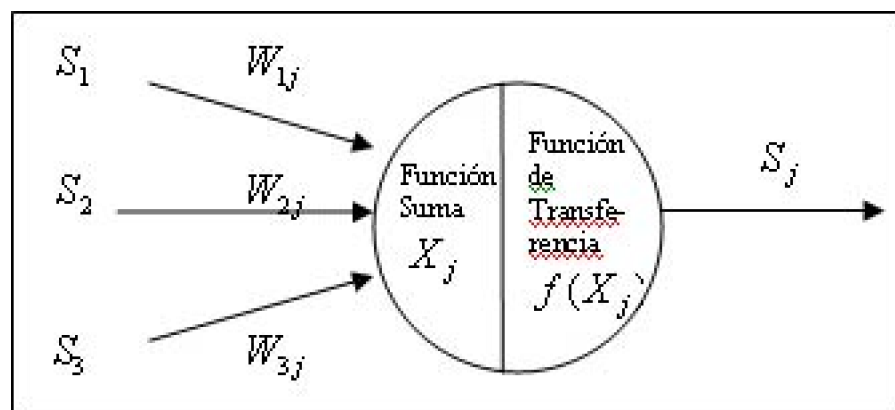


Figura 1: Modelo simple de una Neurona

La estructura clásica de una red neuronal se ve reflejada en la siguiente arquitectura:

¹ [Jiménez 96a]

² [Reyes 02a]

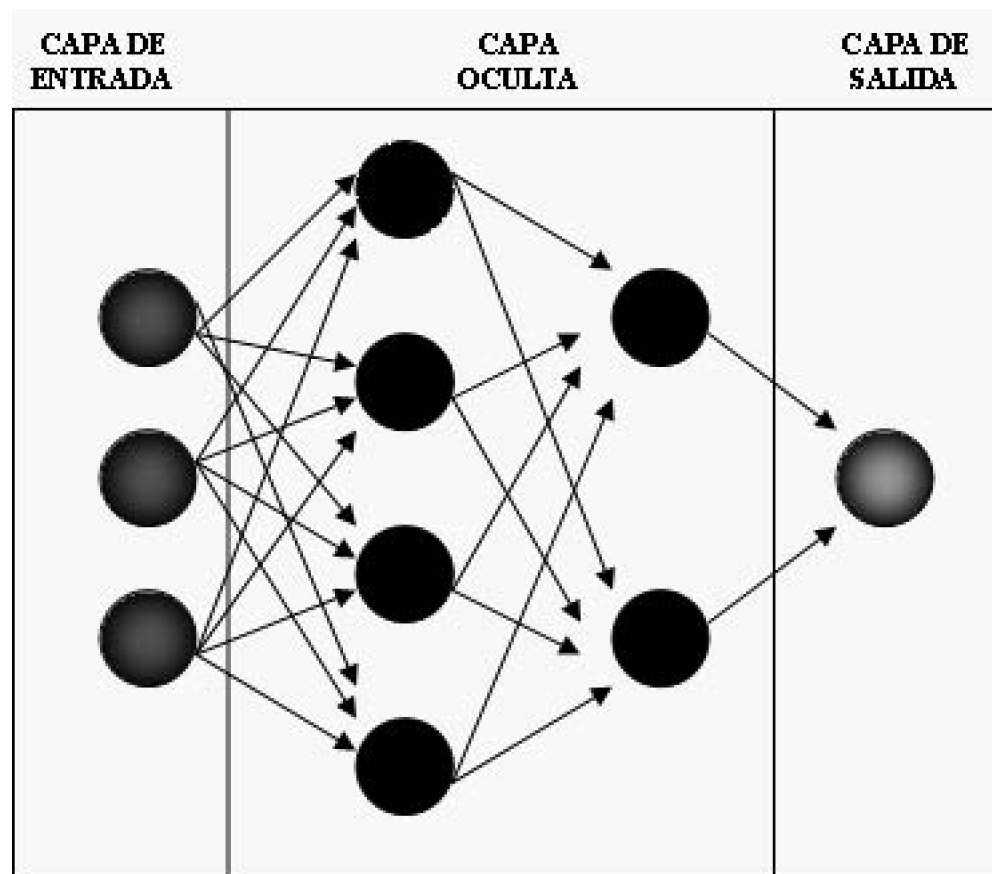


Figura 2: Estructura clásica de una red neuronal

Donde se pueden distinguir como parte de su diseño:

1. Capa de entrada:

Destinada a recibir la información para la resolución del problema (variables independientes). Tiene como características:

- a) Ser de carácter única.
- b) Número de neuronas definido por la cantidad de inputs.
- c) Propaga los datos a la capa siguiente sin proceso alguno, salvo la aplicación de una función escalar.

2. Capa Oculta:

Donde se lleva a efecto el procesamiento de los datos. Tiene como características:

- a) La no cualidad de única, pudiendo existir más de una en la arquitectura de red.
- b) Número de neuronas definido por quien aplica la red.
- c) Los datos se propagan a la capa siguiente previo acto de una función de activación.

3. Capa de Salida:

En las que se presenta el resultado entregado por la red neuronal. Tiene como características:

- Ser de carácter única.
- Número de neuronas definido por la cantidad de outputs.
- Neuronas aplican una función de activación a los datos que reciben, antes de dar el producto final.³

1. Las Neuronas

La neurona, también llamada procesador elemental (PE), es 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 (ver figura 3). Los procesadores elementales se agrupan formando capas, las cuales a su vez conforman las llamadas redes neuronales.

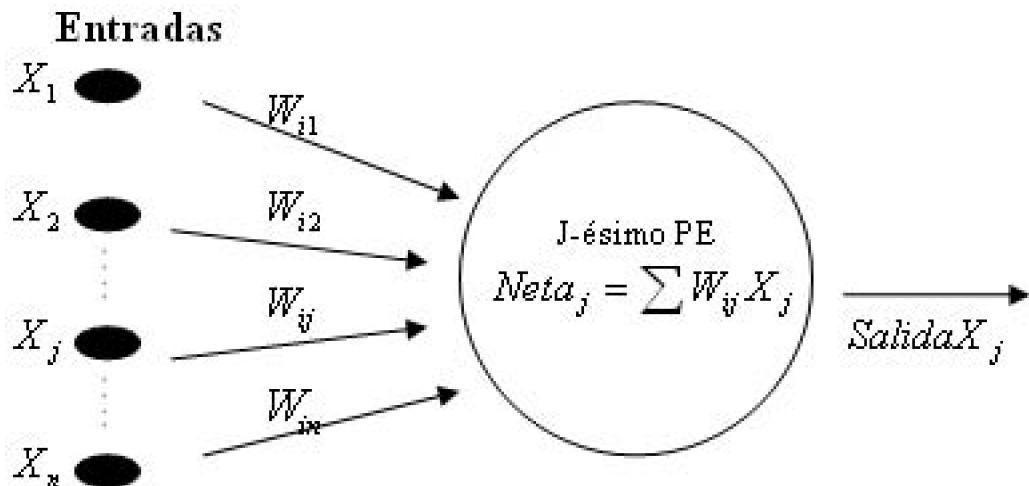


Figura 3: La Neurona

Al igual que una neurona biológica, el PE tiene muchas entradas pero una sola salida, que se puede aplicar a muchos otros PEs de la red. La salida del j-ésimo nodo es X_j , del mismo modo que la salida generada por el i-ésimo nodo se denomina X_i . Cada conexión con el i-ésimo PE tiene asociada a él un valor llamado *peso* o *intensidad de conexión*. El peso de la conexión procedente del j-ésimo nodo y que llega al i-ésimo nodo se denomina W_{ij} . Una conexión de entrada puede ser excitatoria (pesos positivos) o inhibitoria (pesos negativos). Cada PE determina un valor de entrada neto basándose en todas las conexiones de entrada. La forma más común es calcular el valor de entrada neto sumando los valores de entrada, ponderados mediante sus pesos correspondientes. Es decir, la entrada neta de la i-ésima unidad se puede escribir de la siguiente forma:

$$Neta_i = \sum_j X_j W_{ij} \quad (7)$$

³ [EspinozaMu 02a]

En donde el índice j recorre todas las conexiones que posea el PE. Una vez que la entrada neta ha sido calculada, se transforma en el valor de activación para ese PE.

$$a_i(t) = F_i(a_i(t-1), \text{neta}_i(t)) \quad (8)$$

En la mayoría de los casos, la activación y la entrada neta son idénticas, y los términos suelen utilizarse de manera intercambiable.

Una vez que se ha calculado la activación del PE, se puede determinar el valor de salida aplicando la función de salida que puede escribirse como:

$$X_i = f_i(\text{neta}_i) \quad (9)$$

2. Funciones de Activación

Las funciones de activación son los elementos de la topología de la red que determinan en que nivel de datos se quiere poner el mayor énfasis. Por ejemplo, si se utiliza una función lineal todos los datos son tratados de la misma manera; por el contrario, si se usara una función tangente hiperbólica, ésta consideraría más relevante la información alrededor de cero que aquella que se encuentra en los extremos de uno y menos uno.

Los tipos de función de activación que podrían utilizarse son ilimitados. Sin embargo las más comunes (más utilizadas) son las siguientes:

2.1 Función Logística

La función Logística o Sigmoide es la más común en la mayor parte de las redes neuronales con aplicaciones financieras. Su rango es $(0,1)$. Se utiliza para concentrar el aprendizaje en valores no extremos en donde deberían encontrarse la mayor parte de los casos, por ello es de especial utilidad cuando los outputs son categorías. En términos matemáticos su expresión es:

$$F(x) = \frac{1}{(1 + e^{-x})} \quad (10)$$

Figura 4: Función Logística

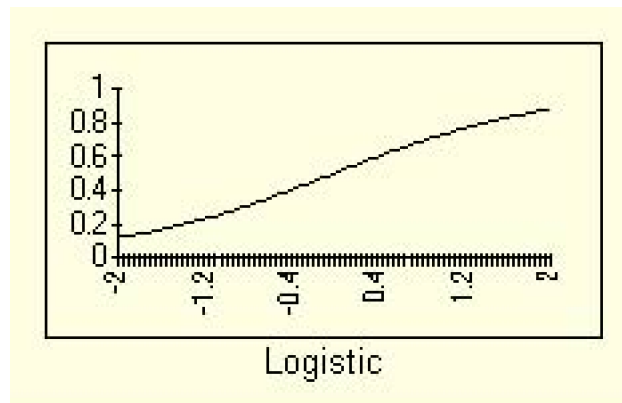


Figura 4: Función Logística

2.2 Función Lineal

La función Lineal es especialmente útil en aquellos casos en que el output es una variable continua o en aquellos en que se desea que la red aprenda los eventos menos frecuentes, es decir, aquellos que coincidirán con valores alejados a cero en la escala. Ello se debe a que a diferencia de la función Sigmoide, la Lineal no se hace menos sensible al alejarse de cero. La expresión matemática es:

$$F(x) = x \quad (11)$$

Figura 5: Función Lineal

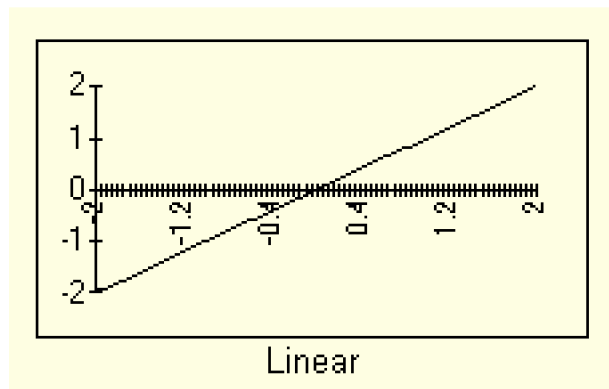


Figura 5: Función Lineal

2.3 Función Tangente Hiperbólica

La Función Tangente Hiperbólica tiene las mismas propiedades que la Logística, sin embargo, el rango de salida de esta función permite respuestas simétricas (-1,1); manteniendo una intermedia en cero. Esto puede ser de especial interés al escalar los datos en la primera capa de la red, pues en muchos casos puede necesitarse atenuar los valores exageradamente altos o bajos, pero diferenciándolos de un nivel donde la información sea irrelevante en donde se tendrá como valor cero.

Esta función suele converger antes que la función Logística, sin embargo, no

necesariamente generaliza igualmente bien. En términos matemáticos la función es:

$$F(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \quad (12)$$

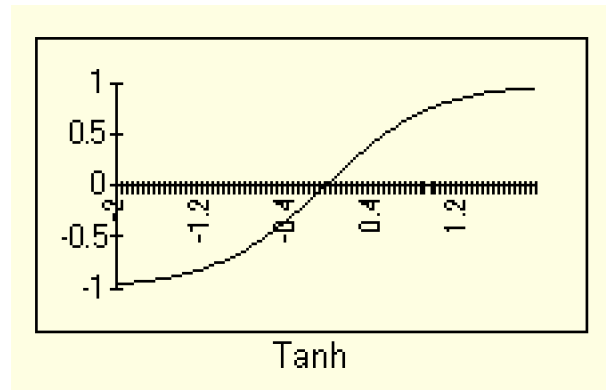


Figura 6: Función Tangente Hiperbólica

2.4 Tangente Hiperbólica 1,5

La Función Tangente Hiperbólica con parámetro 1,5, concentra más el aprendizaje en valores cercanos a cero amplificando el valor de la variable de entrada por 1,5. La diferencia con la Tangente Hiperbólica convencional debe probarse empíricamente. La fórmula es:

$$F(x) = \frac{(e^{1.5x} - e^{-1.5x})}{(e^{1.5x} + e^{-1.5x})} \quad (13)$$

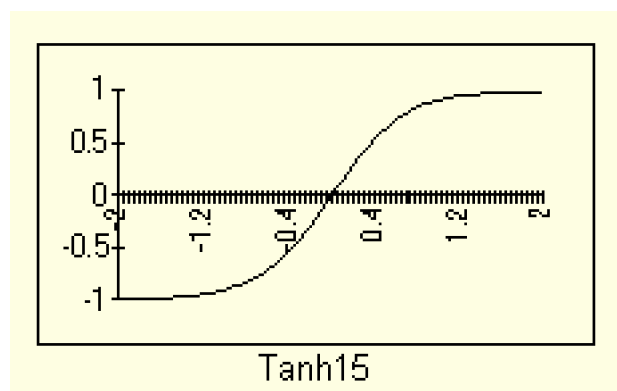


Figura 7: Función Tangente Hiperbólica 1,5

2.5 Seno

El uso de la función Seno es sólo una alternativa adicional a probar, su forma en algunos casos puede implicar la obtención de un mejor ajuste. En el gráfico se observa esta función. Su fórmula es:

$$F(x) = \text{Sen}(x) \quad (14)$$

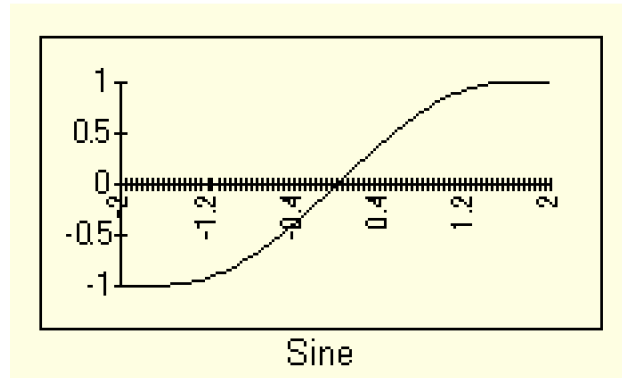


Figura 8: Función Seno

2.6 Logística Simétrica

Esta función es similar a la Sigmoide, pero se mueve en el rango (-1,1). La aplicación de esta función de transferencia debe considerarse de acuerdo al escalamiento de los datos. Su expresión matemática es la siguiente:

$$F(x) = \frac{2}{e^{-x} + 1} - 1 \quad (15)$$

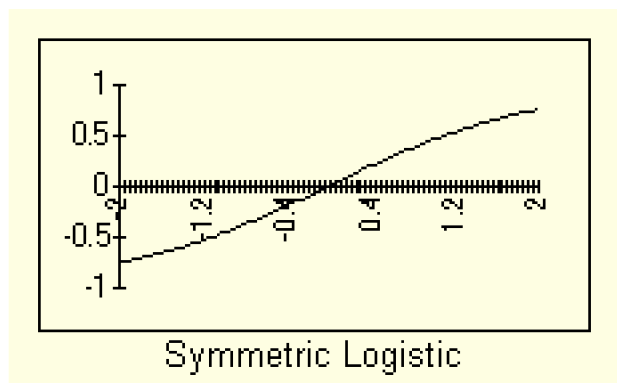


Figura 9: Función Logística Simétrica.

2.7 Gaussiana

La principal diferencia entre la función Gaussiana y las anteriores, está en que ésta no es monótona creciente, se utiliza para extraer información relevante de los extremos. Produce salidas en el rango (0,1) y su fórmula es la siguiente:

$$F(x) = e^{-x^2} \quad (16)$$

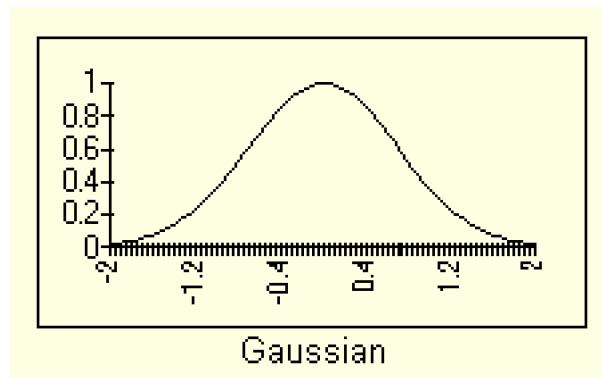


Figura 10: Función Gaussiana

2.8 Gaussiana Complemento

Al igual que la función Gaussiana pretende enfatizar información importante en valores extremos de los datos, con la diferencia de que para estos valores toma el valor de uno en vez de cero como en la Gaussiana (ver gráfico). Estas dos últimas funciones suelen resultar especialmente útiles al utilizarse combinadas con la función Logística o Tangente Hiperbólica en las capas ocultas de la red. Matemáticamente, esta función es:

$$F(x) = 1 - e^{-x^2} \quad (17)$$

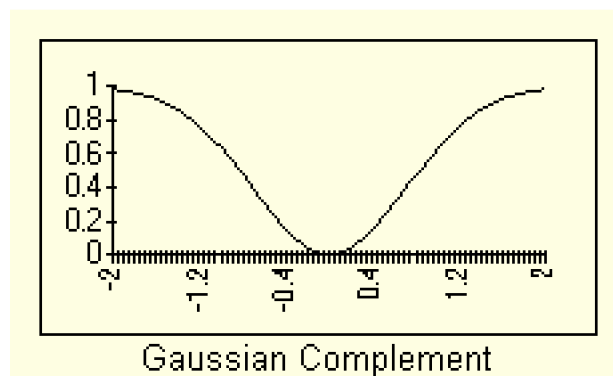


Figura 11: Función Gaussiana Complemento

3. Características de la Redes Neuronales Artificiales (R.N.A.)

Las Redes Neuronales Artificiales cuentan con varios elementos o características que las distinguen de otros sistemas. En primer lugar, su estructura se inspiró en el cerebro humano con el objeto de reproducir alguna de sus capacidades. De esta manera, están formadas por neuronas (artificiales) o elementos de procesamiento (PE), las cuales al igual que su contraparte biológica, pueden comunicarse a través de sinapsis (unión entre dos neuronas). Los elementos de procesamiento se pueden excitar o inhibir (dependiendo de la intensidad de los pesos), aprender, agruparse, etc. Cada conexión entre dos unidades tiene asignado una dirección y un peso. El conocimiento del modelo de la red queda almacenado en dichos pesos. Cada unidad de la capa oculta y de salida

calcula su valor como la suma del producto de los pesos y entradas (función neta), pasando este resultado a través de una función no lineal:

$$f_i(\text{net}_i)^5 \quad (18)$$

A fin de resolver problemas de modo efectivo, donde se apoye la toma de decisiones en base a experiencias pasadas que se asocian y aplican en la generación de conocimiento destinado a nuevas situaciones, las Redes Neuronales se caracterizan por actuar de forma:

a) Paralela: Las R.N.A. cuentan con una gran cantidad de procesadores elementales, cada uno de los cuales trabaja simultáneamente con una parte de un problema mayor. Esta característica resulta de gran utilidad, por ejemplo, en el reconocimiento de imágenes visuales. Un PC que trabaja en secuencia, instrucción a instrucción, tardaría varios minutos en realizar una sencilla tarea de tratamiento sobre una imagen de 256x256 píxeles. Un sistema basado en 16 DSP operando en paralelo emplearía del orden de 20 ms en la misma tarea. En cambio, el cerebro tarda ese mismo tiempo en procesar millones de píxeles. La clave reside en que en este último caso los miles de millones de neuronas que intervienen en el proceso de visión están operando en paralelo sobre la totalidad de la imagen.⁴

b) Distribuida: Los cálculos asignados se distribuyen entre todas las neuronas, cada una de las cuales trata una fracción de un problema mayor, a diferencia de los computadores convencionales que sólo cuentan con un procesador (CPU) y memoria fija.

c) Adaptativa: Las R.N.A. tienen la capacidad de adaptarse al entorno modificando sus pesos y sinapsis, aprenden de la experiencia, pudiendo generalizar conceptos a partir de casos particulares y así poder encontrar una solución aceptable al problema.

4. Clasificación de las Redes Neuronales

Las Redes Neuronales se pueden clasificar de acuerdo a su aprendizaje, arquitectura o tipo de resultado. A continuación expondremos las distintas clasificaciones.

4.1 Clasificación de acuerdo a su aprendizaje

4.1.1 Supervisados: Consiste en presentarle un conjunto de patrones a la red, junto con la salida deseada u objetivo, la red es capaz de ajustar los pesos de las neuronas hasta que su salida tiende a ser la deseada, utilizando para ello información detallada del error que comete en cada paso. De esta forma, la red es capaz de estimar relaciones entrada/salida sin necesidad de proponer una cierta forma funcional de partida.

4.1.2 No Supervisados: Se le presentan a la red una multitud de patrones, sin adjuntar la respuesta que deseamos. Así, durante el proceso de aprendizaje, la red autoorganizada debe descubrir por sí misma rasgos comunes, regularidades, correlaciones o categorías en los datos de entrada, e incorporarlas a su estructura interna

⁴ [Del Brío 97a]

de conexiones (pesos). Se dice, por tanto, que las neuronas deben autoorganizarse en función de estímulos (señales o datos) procedentes del exterior. Para obtener resultados de calidad la red requiere de un cierto nivel de redundancia en las entradas procedente del espacio sensorial; en definitiva, un número de patrones de aprendizaje suficiente.

4.1.3 Híbridos: Combinación de los dos tipos básicos de aprendizaje en la red, normalmente en distintas capas de neuronas.

4.1.4 Reforzados: Se encuentran entre el aprendizaje supervisado y no supervisado. Como el supervisado, emplea información sobre el error cometido, pero solamente indicando lo bien o lo mal que se está actuando. Como en el caso no supervisado, no suministra explícitamente la salida deseada.⁵

4.2 Clasificación según su arquitectura

4.2.1 Realimentadas: La información puede circular entre las capas en cualquier sentido, incluido el de salida-entrada. En este caso, se tiene un conjunto de neuronas que actúa como memoria de las características de los patrones procesados previamente.

4.2.2 Unidireccionales: La información fluye en un solo sentido, desde las neuronas de entrada hacia las de salida. El proceso de ajuste se lleva a cabo por épocas, existiendo un ajuste sólo en términos del error obtenido al comparar con el resultado real.

4.3 Clasificación según tipo de resultado

4.3.1 Pronóstico o Estimación en Nivel: El output es un valor, una cantidad. Corresponde a una variable continua.

4.3.2 Clasificación: El output es una probabilidad, entrega categorías. Corresponde a una variable discreta.

5. Tipos de funcionamiento de las Redes Neuronales Artificiales

5.1 Aprendizaje o Entrenamiento

Cuando se construye una red neuronal, se parte de un modelo de neuronas y de una determinada arquitectura de red, estableciéndose los pesos iniciales como nulos o aleatorios. Para que la red resulte útil es necesario entrenarla, lo que se conoce como modo de aprendizaje. Aprendizaje es el proceso por el cual se produce el ajuste de los parámetros libres de la red (pesos) a partir de un proceso de estimulación por el entorno que rodea la red. En términos más simples significa la extracción de regularidades de los datos de entrenamiento que serán transferidos a nuevos ejemplos. Esto se hace a través de una cierta regla de aprendizaje (manera sistemática de cambiar los pesos), construido normalmente a partir de la optimización de una función de error o costo, que mide la eficacia actual de la operación de la red. En un proceso de aprendizaje la información

⁵ [Reyes 02d]

contenida en los datos de entrada queda incorporada en la propia estructura de la red neuronal, estructura que almacena la representación de una cierta imagen de su entorno.

El modo de aprendizaje es de gran importancia, debido a que una característica esencial de las R.N.A. es que son sistemas entrenables, capaces de realizar un determinado tipo de procesamiento, aprendiéndolo a partir de un conjunto de ejemplos.

Hay básicamente tres aspectos relacionados con el aprendizaje: La elección del conjunto de entrenamiento y su tamaño, la elección de los parámetros del aprendizaje, y el momento en que debemos detener el aprendizaje. Para enfrentar esto sólo existen algunas reglas generales, de manera que la *experimentación* juega un papel importante. En este aspecto, la disponibilidad de ambientes de simulación rápidos y habilidades de prueba extensivos son una gran ventaja.⁶

5.2 Regla de Retropropagación

Existen diferentes algoritmos para entrenar las redes, siendo fundamental el de Retropropagación del Error, Back-propagation o Regla del Gradiente. Este algoritmo permite modificar los pesos de cada una de las neuronas de la red en forma proporcional al error de su salida (generalmente la diferencia entre lo deseado y lo obtenido), el cual por medio de la regla se va propagando desde la salida final por las capas intermedias hasta la entrada. Los pasos a seguir para este entrenamiento son los siguientes:

- a) Aplicar un vector de entrada como estímulo para la capa de entrada.
- b) Propagar esta señal por todas las capas de la red hasta obtener una salida.
- c) Comparar la salida de la red con el output deseado.
- d) Calcular una señal de error para cada unidad de salida.
- e) Transmitir la señal de error hacia atrás hacia todos los nodos que estén conectados directamente.
- f) Los nodos de capas anteriores reciben una fracción del error proporcional al peso de su conexión.
- g) Actualizar iterativamente los pesos de todas las conexiones de tal manera de disminuir el error.
- h) En el caso de que no se tenga convergencia, repetir todo lo anterior. Cuando el error resulta aceptablemente pequeño, el aprendizaje se da por concluido.⁷

5.3 Elección del conjunto de entrenamiento

El tamaño del conjunto de entrenamiento es de vital importancia para la utilidad práctica de la red. Si los patrones de entrenamiento no transfieren todas las características del problema, el mapeo descubierto durante el entrenamiento sólo se aplica al conjunto de entrenamiento. Así, el rendimiento en el conjunto de prueba será mucho peor que el

⁶ [Reyes 02e]

⁷ [Virreira 96d]

rendimiento del conjunto de entrenamiento. La única regla general es usar muchos datos y representativos, es decir, datos que cubran todos los escenarios posibles con los que se pueda encontrar la red en el futuro.

La relación entre el tamaño y el conjunto de entrenamiento y el número de pesos de la red neuronal juega un papel importante. Si el número de ejemplos de entrenamiento es menor que el número de pesos se podría producir una memorización de la solución, es decir, que la red almacene cada ejemplo del conjunto de entrenamiento, lo que se traducirá en una pobre generalización (capacidad de proporcionar una respuesta correcta ante datos no utilizados en su entrenamiento). Es recomendable que el número de muestras de entrenamiento sea de al menos el doble que el número de pesos de la red.

Cuando hay una gran discrepancia entre el rendimiento del conjunto de entrenamiento y el de prueba, se debe detener el aprendizaje. En estos casos se debe aumentar el tamaño del conjunto de entrenamiento y/o producir una mezcla diferente de ejemplos de entrenamiento y prueba. Siempre se puede esperar una caída en el rendimiento (más o menos entre 10% a 15%) desde el conjunto de entrenamiento al de prueba.⁸

5.4 Tamaño de la red

La forma más eficiente para determinar el tamaño óptimo de la red es a través de la *experimentación*. Un parámetro importante es el número 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 su capacidad para memorizar el conjunto de entrenamiento. Sin embargo, si se continúa aumentando el tamaño de la red, hay un punto donde la generalización empeorará, debido al hecho que se puede sobreajustar el conjunto de entrenamiento, de modo que cuando la red trabaja con patrones que nunca ha visto anteriormente la respuesta es impredecible. El problema es encontrar cuál es la combinación correcta de neuronas y número de capas para resolver el problema en tiempos de entrenamiento más bajo. Cada capa aumenta el poder discriminante de la red. Es recomendable comenzar con una red pequeña y observar el comportamiento de la curva de aprendizaje. Si el error final es pequeño, el número de procesadores elementales probablemente es el apropiado. Si el error final es grande, el aprendizaje puede haber caído en un mínimo local (con lo cual el número de neuronas debe ser aumentado). En el caso que el rendimiento de la red se deteriore considerablemente desde el conjunto de entrenamiento al conjunto de prueba, una de dos cosas ha ocurrido: el conjunto de entrenamiento no es representativo del dominio del problema, o se ha configurado la red con muchos pesos produciendo una pobre generalización (sobreajuste).

Un método alternativo para el tamaño de la red, es comenzar con una red grande e ir cambiando alguno de los pesos. Los pesos pequeños que tiendan a cero y decaen de iteración a iteración, serán eliminados.⁹

⁸ [Reyes 02f]

⁹ [Reyes 02g]

5.5 Parámetros de aprendizaje

No existe una recomendación o receta en lo que se refiere a parámetros de aprendizaje. El objetivo es entrenar tan rápido como sea posible y alcanzar el mejor rendimiento. Aumentando el parámetro de la tasa de aprendizaje, se disminuirá el tiempo de entrenamiento, pero también aumentarán las posibilidades de divergencia, es decir, de alejarse del valor óptimo. Debido a que la corrección del peso depende de las características de la superficie de error y de la tasa de aprendizaje, para obtener un aprendizaje constante es necesario un parámetro adaptativo. Se recomienda una estrategia en donde la tasa de aprendizaje sea grande al comienzo del proceso de aprendizaje y que progresivamente decaiga hacia el término de la adaptación.

Las actualizaciones pueden ejecutarse al término de la presentación de todos los elementos del conjunto de entrenamiento o en cada iteración (tiempo real). La primera modalidad suaviza el gradiente y puede dar un aprendizaje más rápido para datos ruidosos, sin embargo, también puede promediar los gradientes en cero y evitar el aprendizaje. La modificación de los pesos en cada iteración con una tasa de aprendizaje pequeña puede ser preferible la mayor parte del tiempo.

La curva del ECM (error cuadrático medio) como función del número de iteraciones es llamada *curva de aprendizaje*. Cuando la curva de aprendizaje presenta una forma horizontal, el tamaño del paso (parámetro que determina la magnitud en la cual se van a actualizar los pesos) debería ser incrementado para acelerar el aprendizaje y mejorar su rendimiento. Por otro lado, cuando la curva de aprendizaje presenta una forma oscilatoria el tamaño del paso debiera ser disminuido. En un caso extremo, el error aumenta uniformemente, mostrando que el aprendizaje es inestable. En este punto la red debería ser reformulada. Cuando la curva de aprendizaje se estabiliza después de muchas iteraciones en un nivel que no es aceptable, es tiempo de cambiar la topología de la red (más neuronas en la capa oculta o más capas ocultas, o una topología diferente) o el procedimiento de entrenamiento.¹⁰

5.6 Criterio de detención

Otro problema es cuándo detener el entrenamiento. Todos los criterios están basados en el error cuadrático medio. Los criterios más utilizados son fijar el número de iteraciones o el prefijar un error final. Lo que se usa en la práctica es el aprendizaje mínimo incremental (cuando entre dos iteraciones consecutivas el error no disminuye en al menos una cantidad dada o umbral, el entrenamiento debiera terminarse). Esto nos da un criterio para comparar topologías muy diferentes. Otra alternativa es usar dos criterios para detener el entrenamiento: fijar un ECM para el conjunto de prueba o fijar un número máximo de eventos (es la presentación de un solo patrón de entrenamiento a la red) desde que se alcanzó el último mínimo. El aprendizaje se detiene cualquiera de las dos alternativas anteriores es alcanzada.¹¹

¹⁰ [Reyes 02h]

¹¹ [Reyes 02i]

5.7 Memorización v/s Generalización

Uno de los aspectos fundamentales se relaciona con la capacidad de las redes neuronales de generalizar a partir de ejemplos, lo que se denomina en ocasiones el problema de la *memorización* frente a la *generalización*. La Generalización es la capacidad de proporcionar una respuesta correcta ante patrones que no han sido empleados en su entrenamiento. En su proceso de entrenamiento se puede considerar, por una parte, un *error en aprendizaje* y por otra un *error en generalización* (error en test). Una red se debe entrenar hasta que su *error de generalización* sea mínimo.

Para medir de una forma objetiva la eficacia final del sistema construido, el conjunto de patrones de partida debería ser dividido en tres grupos, un conjunto de *aprendizaje* y dos de test, uno para la realización de *validación cruzada* (que evite el sobreajuste) y otro que se mantenga al margen del proceso de aprendizaje, permitiendo de este modo una validación totalmente objetiva de la red neuronal desarrollada.

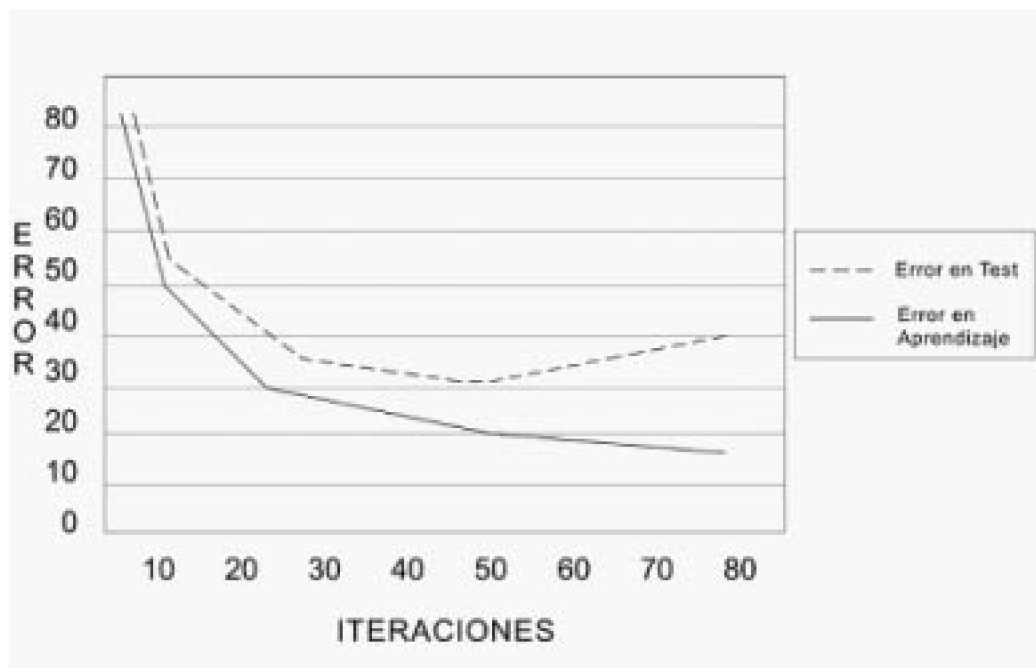


Gráfico 1: Error en Test v/s error en Aprendizaje

5.8 Recuerdo o Ejecución

Una vez terminado el aprendizaje, los pesos y la estructura quedan fijos, la red neuronal está en condiciones de procesar datos. Esto es lo que se conoce como *modo recuerdo o ejecución*.

6. Tipos de Redes Neuronales Artificiales

6.1. Perceptrón Multicapa (MLP)

Son redes unidireccionales entrenadas con el algoritmo de propagación hacia atrás (BP). Son redes con aprendizaje supervisado, de manera que requieren una respuesta deseada para ser entrenadas. Son ampliamente usados para clasificar patrones. Usan eficientemente la información contenida en los datos de entrada. Estos modelos actúan como funciones mapeadoras universales, desempeñándose muy bien con las series de tiempo.¹²

La red aprende un conjunto predefinido de pares de entrada y salida dados como ejemplos, empleando un ciclo de 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 hacia 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 cambiario.

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. Como señalamos anteriormente, 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, de acuerdo a Leung, Daouk y Chen (2000), los modelos propuestos para predecir el signo 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.¹³

6.1.1 Ventajas del MLP

a) Se puede usar en una multitud de problemas diferentes, proporcionando con frecuencia buenas soluciones con no demasiado tiempo de desarrollo.

b) Con una o dos capas ocultas pueden aproximar virtualmente cualquier mapa de entrada-salida.

6.1.2 Desventajas del MLP

¹² [Parisi 02a]

¹³ [Parisi 02b]

a) Su lentitud de convergencia. Para este inconveniente existe una variante a modo de corrección del problema. Se trata de incluir un término adicional al algoritmo de aprendizaje (BP), el denominado término de *momento*, que consiste en añadir al cálculo de la variación de los pesos, un término adicional proporcional al incremento de la iteración anterior. Si el incremento en un determinado peso es alto, en la siguiente iteración lo será todavía más. Por otra parte, si los incrementos en un determinado peso oscilan (a veces son positivos, otras negativos), el incremento efectivo acumulado se reduce al cancelarse. Esta es una manera de aumentar el ritmo de aprendizaje efectivo en determinadas direcciones.

Puede incurrir en sobreajuste.

b) No garantiza alcanzar el mínimo global de la función de error.

c) Son clasificadores estáticos, es decir, el mapa de entrada-salida depende sólo de la entrada actual.

d) Necesita un gran conjunto de datos de entrada.

6.2 Variantes del MLP

6.2.1 Conexión Jump

En este tipo de red de retropropagación, cada capa está conectada a la o las capas anteriores, es decir, por ejemplo en una red de tres capas, la capa oculta está conectada a la de entrada y la capa de salida lo está con la de entrada y la oculta. Esta arquitectura puede ser útil cuando se está trabajando con problemas muy complejos, es decir, cuando puede ser muy difícil para una persona definir los diferentes patrones que son inherentes a los datos.

6.2.2 MLP con Conexiones Recurrentes

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 que la red BP estándar, excepto que 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) extra 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.¹⁴

Existen 3 tipos de redes MLP con conexiones recurrentes:

a) La capa de entrada realimentada en la capa de entrada: el término largo de memoria (“slab extra”), recuerda los nuevos datos de entrada y los usa cuando el próximo patrón es procesado.

¹⁴ [Parisi 02c]

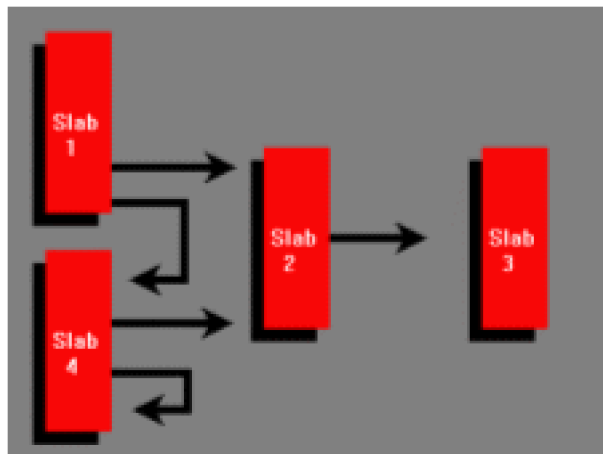


Figura 12: Capa de entrada realimentada en la capa de entrada

b) La capa oculta realimentada en la capa de entrada: el término largo de memoria recuerda la capa oculta, la cual contiene las características detectadas en los datos de patrones anteriores. Esta es la red recurrente más poderosa.

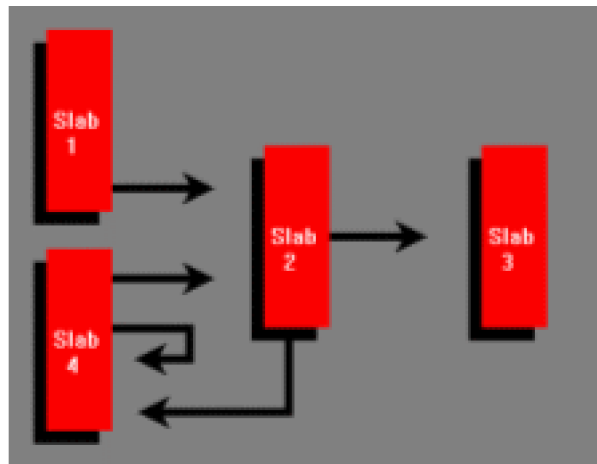


Figura 13: La capa oculta realimentada en la capa de entrada

c) La capa de salida realimentada a la capa de entrada: el término largo de memoria recuerda la salida previamente predicha.

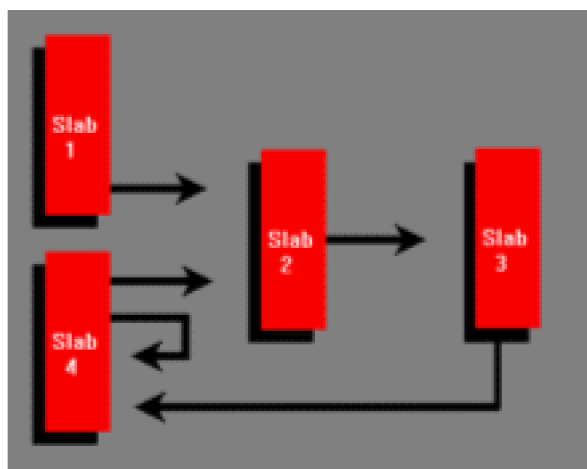


Figura 14: Capa de salida realimentada a la capa de entrada

6.2.3 Red Ward

La red Ward se caracteriza porque pueden aplicarse diferentes funciones de activación a los *slabs* 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.

Existen 3 tipos de red Ward:

a) Dos *slabs* ocultos con diferentes funciones de activación: es una red BP regular de tres capas con dos *slabs* en la capa oculta. Usa una función de activación diferente para cada *slab* en la capa oculta para detectar diferentes características en los datos.

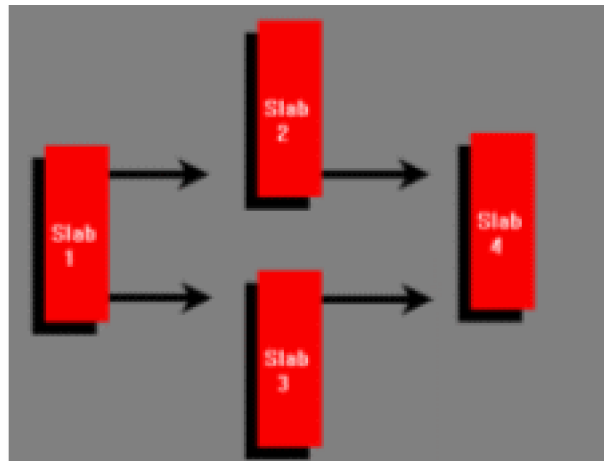


Figura 15: Dos *slabs* ocultos con diferentes funciones de activación

b) Tres *slabs* ocultos con diferentes funciones de activación: es una red BP que agrega un tercer *slab* a la capa oculta. Cuando cada *slab* en la capa oculta tiene una función de activación diferente, ofrece tres formas diferentes de ver los datos.

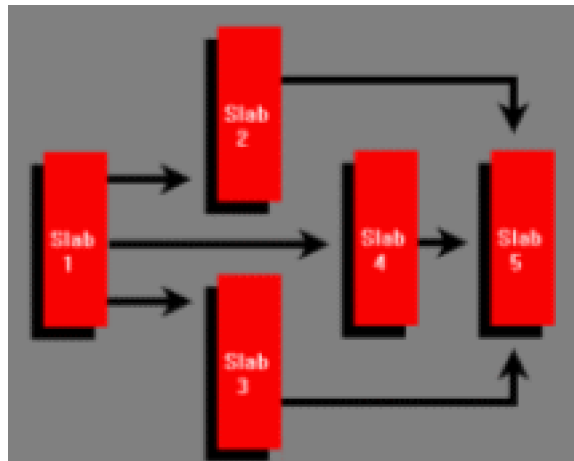


Figura 16: Tres slabs ocultos con diferentes funciones de activación

c) Dos *slabs* ocultos, diferentes funciones de activación más una conexión Jump: esta red Ward es un BP regular con dos *slabs* en la capa oculta y una conexión Jump entre la capa de entrada y la capa de salida. La capa de salida recibe dos vistas diferentes de los rasgos de los datos como descubrió en los *slabs* ocultos más las entradas originales.

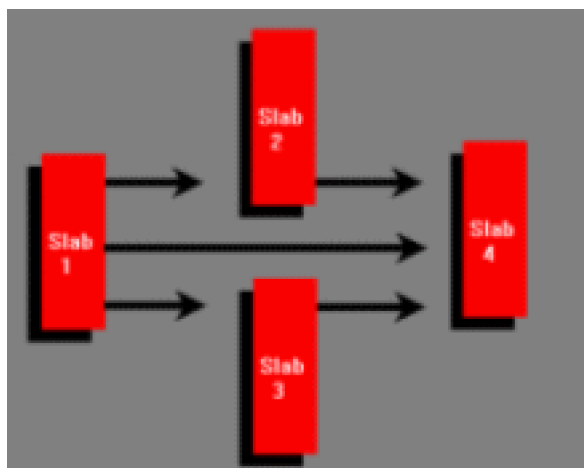


Figura 17: Dos slabs ocultos, diferentes funciones de activación más una conexión Jump

6.3 Función de Base Radial

Es un modelo híbrido que incorpora tanto aprendizaje supervisado como no supervisado. En su estructura cuenta con 3 capas de neuronas:

a) Capa de entrada: Donde las neuronas simplemente envían la información del exterior hacia las neuronas de la capa oculta.

b) Capa oculta: Las neuronas de la capa oculta operan en base a la distancia que separa el vector de entrada respecto del vector sináptico que cada una almacena (denominado Centroide) y su respuesta es localizada, pues sólo responden con una intensidad apreciable cuando el vector de entradas presentado y el Centroide de la neurona pertenecen a una zona próxima en el espacio de las entradas.

c) Capa de salida: En esta capa, las neuronas son lineales, esencialmente calculan la suma ponderada de las salidas que proporciona la capa oculta.

6.3.1 Ventaja

Encuentra la entrada al mapa de salida usando aproximadores locales. Se entrena rápidamente y requiere pocas muestras de entrenamiento.

6.3.2 Desventaja

Un problema es que puede requerir un espacio dimensional muy grande para solucionar un problema.

6.4 Análisis de Componentes Principales (PCA)

Es una mezcla de aprendizaje supervisado y no supervisado. El PCA es un procedimiento lineal para encontrar la dirección en el espacio de entrada en donde se produce la mayor parte de la energía de la entrada. En otras palabras, el PCA ejecuta una extracción de características. Las proyecciones de esos componentes corresponden a los valores propios de la matriz Covarianza de Entrada. El segmento no supervisado de la red ejecuta la extracción de características y el segmento supervisado de la red ejecuta la clasificación (lineal o no) de esas características usando un MLP.

Esta técnica transforma las muestras de entrada en un nuevo espacio de modo que la información acerca de las muestras es mantenida, pero la dimensionalidad reducida. Esto hace que la clasificación trabaje en forma más fácil.

La importancia del análisis PCA es que el número de entradas para el MLP puede ser significativamente reducido. Esto conlleva a una reducción del número de patrones de entrenamiento requerido y una reducción en el tiempo de entrenamiento.¹⁵

6.4.1 Ventajas

a) Converge rápidamente.

b) El segmento no supervisado provee un medio de reducción de datos.

6.4.2 Desventaja

Posibilidad de perder importante información de entrada.

6.5 Mapas Autoorganizados de Características (SOFM)

En este modelo, las neuronas se organizan en una arquitectura unidireccional de dos capas. La primera es la capa de Entrada o Sensorial, la cual distribuye la información procedente del espacio de entrada a las neuronas de la segunda capa, en donde se lleva a cabo el procesamiento, formando así el mapa de rasgos. Es una red no supervisada con comportamiento "competitivo".

¹⁵ [Reyes 02]

En el modo de operación normal de la red, los pesos permanecen fijos. En primer lugar, cada neurona (i,j) calcula la similitud entre el vector de entradas X , y su propio vector de pesos sinápticos W_{ij} , según una cierta medida de distancia o criterio de similitud establecido. A continuación, se declara vencedora la neurona g cuyo vector de pesos W_g es más similar al de entradas.

En la fase de *aprendizaje* cada neurona del mapa sintoniza con diferentes rasgos del espacio de entrada. El proceso es el siguiente. Después de la presentación y procesamiento de un vector de entrada $X(t)$, la neurona vencedora modifica sus pesos de manera que se parezcan un poco más a $X(t)$. De este modo, ante el mismo patrón de entrada, dicha neurona responderá en el futuro todavía con más intensidad.

Lo descrito hasta el momento responde a un esquema competitivo clásico, en el que cada neurona actúa en solitario. Sin embargo, el modelo de mapa de Kohonen aporta una importante novedad, ya que incorpora a este esquema relaciones entre las neuronas próximas al mapa. Para ello, introduce una denominada *función de vecindad*, que define un entorno alrededor de la neurona ganadora actual (vecindad). Su efecto es que durante el aprendizaje se actualizan tanto los pesos de la vencedora como los de las neuronas pertenecientes a su vecindad. De esta manera, en el modelo de SOFM se logra que neuronas próximas sintonicen con patrones similares.¹⁶

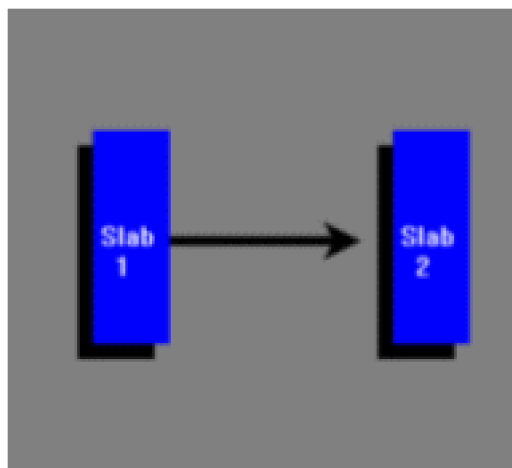


Figura 18: Red SOFM

La utilización de la función vecindad en el modelo de mapas auto-organizados aporta en comparación al modelo competitivo sencillo dos ventajas adicionales: el ritmo efectivo de convergencia se mejora y el sistema es más fuerte frente a variaciones en los valores iniciales de los pesos.

La principal novedad de los SOFM consiste en que la modificación de los pesos no sólo se aplica a una neurona específica (la ganadora), sino también a su vecindad. Así, el proceso de aprendizaje comprende dos fases fundamentales: una ordenación global, en la que se produce el despliegue del mapa, y un ajuste fino en el que las neuronas se especializan.

¹⁶ [Larraín 90a]

6.5.1 Ventajas

- a) Reduce el espacio de entrada en características representativas usando un proceso de autoorganización, de manera que el espacio de entrada es mantenido, mientras que la dimensionalidad del espacio es reducido.
- b) Tiene la capacidad de agrupar y clasificar la información.

6.5.2 Desventaja

Es difícil medir su rendimiento, lo que puede afectar la estabilidad del mapa.

6.6 Time Lagged Recurrent Network (Redes Recurrentes con Retardo Temporal, TLRN)

Son MLP extendidos con conexiones recurrentes locales. Es un modelo muy apropiado para el procesamiento de información temporal, la predicción de series de tiempo, sistemas de identificación y reconocimiento de patrones temporales. El algoritmo de entrenamiento usado es el *backpropagation through time* (BPTT), mecanismo de aprendizaje dinámico que permite especificar ejemplares compuestos por una cantidad de muestras equivalentes a una secuencia de tiempo. Estas muestras son usadas para retropropagar el error a través del tiempo.

Los tipos de memoria disponibles para el TLRN son:

- a) Tap Delay Neuronal Networks (TDNN): Consta de una gran capacidad para imitar el comportamiento de la serie. La memorización alcanza un gran grado de detalle. La estructura de memoria TDNN es simplemente una cascada de retardos ideales (retardos de una muestra).
- b) Gamma: Provee una memoria recursiva de las señales de entradas pasadas.
- c) Laguerre: Es ligeramente más sofisticada que la gamma, ya que ortogonaliza el espacio de memoria, lo cual es útil cuando la red trabaja con núcleos grandes de memoria.¹⁷

6.6.1 Ventajas

- a) Requiere una red pequeña para aprender problemas temporales.
- b) Posee una menor sensibilidad al ruido.

6.6.2 Desventajas

- a) El entrenamiento puede caer en un mínimo local.
- b) El BPTT es muy complejo y requiere mucha memoria.

¹⁷ [Jiménez 96b]

6.7 Red Neuronal Probabilística (PNN)

Estas redes son conocidas por su habilidad para entrenar rápidamente un conjunto amplio de datos. La PNN separa datos en un número especificado de categorías de salidas.

La red PNN trabaja comparando el nuevo patrón de entrada y cada uno de los patrones de entrenamiento de acuerdo a su distancia.

La PNN es una red de tres capas en donde los patrones de entrenamiento son presentados a la capa de entrada y la capa de salida tiene una neurona para cada posible categoría. Deben existir tantas neuronas en la capa oculta como patrones de entrenamiento.

La red produce activaciones en la capa de salida correspondiente a la estimación de la función de densidad de probabilidad para esa categoría. La salida más alta representa la categoría más probable.

El número de neuronas en la capa de entrada es el número de entradas en el problema, y el número de neuronas en la capa de salida corresponde al número de categorías. Debido a que el propósito de una red PNN es separar salidas en categorías diferentes, dos o más salidas son requeridas. Los valores de salida son cero o uno.

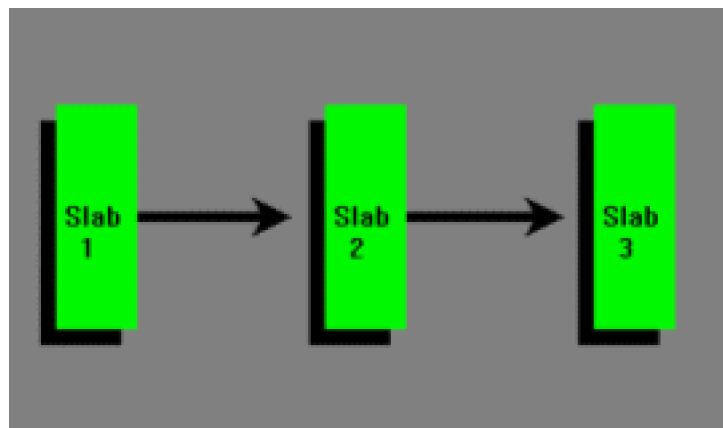


Figura 19: Red Neuronal Probabilística

6.8 Red Neuronal de Regresión General (GRNN)

Estas redes son conocidas por su habilidad para entrenar rápidamente un conjunto amplio de datos. Es un tipo de red supervisada. La GRNN trabaja midiendo cuan lejos un patrón de muestra dado está de los patrones en el conjunto de entrenamiento en un espacio dimensional N , donde N es el número de entradas en el problema. Cuando un nuevo patrón es presentado a la red, ese patrón de entrada es comparado en un espacio dimensional N a todos los patrones en el conjunto de entrenamiento para determinar cuan lejos en distancia está de aquellos patrones. La salida que es predecida por la red es una cantidad proporcional de todas las salidas en el conjunto de entrenamiento. La proporción está basada en cuan lejos el nuevo patrón está de los patrones dados en el conjunto de entrenamiento.

Al igual que la red PNN categoriza datos, pero tiene la capacidad para producir valores de salida continua. Es especialmente útil para funciones de aproximación continuas. La GRNN puede tener entradas multidimensionales, y ajustará un espacio multidimensional a través de los datos.

La GRNN es una red de tres capas donde debe existir una neurona oculta para cada patrón de entrenamiento. No hay parámetros de entrenamiento tales como tasa de aprendizaje y momentos como en el BP, pero hay un factor *smoothing* que es aplicado después que la red es entrenada.¹⁸

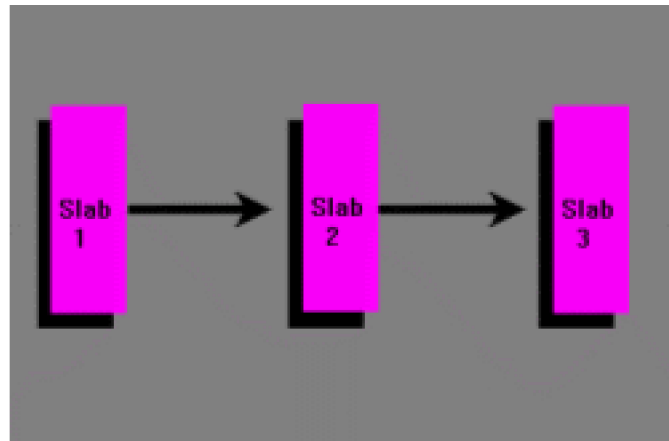


Figura 20: Red Neuronal de Regresión General (GRNN)

6.9 Red GMDH (Group Method of Data Handling)

La red GMDH trabaja construyendo capas sucesivas con uniones (o conexiones) complejas que son los términos individuales de un polinomio. Estos términos polinomiales son creados usando regresión lineal y no lineal. La capa inicial simplemente es la capa de entrada. La primera capa creada es hecha calculando regresiones de los valores en la primera capa junto con las variables de la entrada. De nuevo, sólo los mejores son escogidos por el algoritmo. Éstos se llaman *sobrevivientes*. Este proceso continúa hasta que la red se detiene obteniendo los mejores (según un criterio de selección especificado).

La red resultante puede representarse como un polinomio complejo (es decir, una fórmula familiar). Se puede ver la fórmula que contiene las variables de la entrada más significativas. En algunos aspectos, es como usar análisis de la regresión, pero es más poderoso que el análisis de la regresión. La red GMDH puede construir modelos muy complejos evitando problemas de sobreajuste.

La red GMDH contiene varios métodos de evaluación, llamados criterios de selección, para determinar cuándo debiera terminar el entrenamiento.

Una ventaja de la red GMDH es que reconoce las variables más significativas a medida que entrena, y desplegará una lista de ellas.

¹⁸ [Reyes 02k]

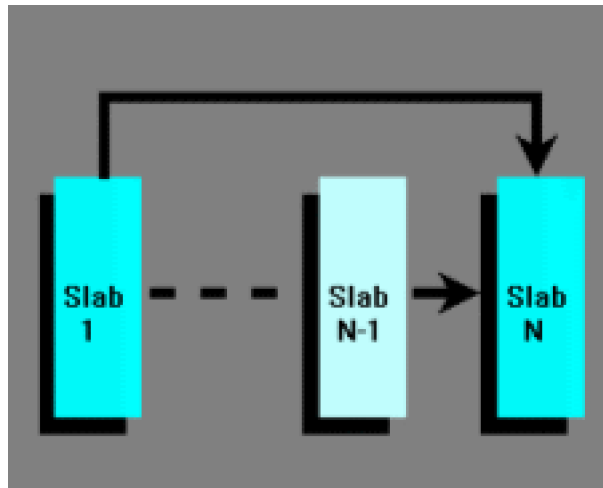


Figura 21: Red GMDH (Group Method of Data Handling)

7. Ventajas de las Redes Neuronales Artificiales

Las mayores ventajas de procesamiento de los sistemas de Redes Neuronales Artificiales son:

a) Adaptabilidad: Esta es la capacidad de las redes neuronales de aprender cómo realizar ciertas tareas al presentarle ejemplos ilustrativos. Si es necesario, una red neuronal ya entrenada y operando puede realizar un aprendizaje continuo presentándole constantemente nuevos ejemplos. En este sentido, las redes neuronales pueden mejorar su capacidad de discriminar entre entradas similares o adaptarse a cambios del ambiente.

b) Tolerancia a fallas: Si una neurona individual o una porción completa de la red neuronal es destruida, continúa funcionando adecuadamente. Cuando el daño llega a ser tan extensivo que el comportamiento de la red comienza a ser afectado, el efecto es una degradación gradual de rendimiento, en forma opuesta a una falla completa.

La razón por la cual las redes neuronales son capaces de tener este tipo de tolerancia a fallas, es que ellas almacenan la información de una manera distribuida. Una pieza de conocimiento o información no es codificada en una sola neurona sino que sobre varias. Esto contrasta con las tecnologías tradicionales de procesamiento de la información, en las cuales cada elemento de información es almacenado en una localización específica de memoria. En estos casos, un pequeño mal funcionamiento de *hardware* puede causar pérdidas catastróficas de información y fallas de programas.

c) Autoorganización: Durante el entrenamiento, una red neuronal organiza y crea su propia representación de la información que recibe. Esto es algo diferente a las metodologías tradicionales de programación, las cuales requieren que toda la información sea especificada explícitamente por el programador. Consecuentemente, las redes neuronales son muy adecuadas para problemas en los cuales es difícil o imposible definir un modelo explícito, programas o reglas para obtener la solución. Esta es una ventaja clara, ya que existe un gran número de problemas de este tipo.

Un resultado útil de esta capacidad de autoorganización es que la red neuronal logra la habilidad de generalización. Una red neuronal entrenada para reconocer ciertos patrones creará su propia categoría de patrones levemente diferente o poco familiar intentará asociarlo con alguna de las categorías de patrones que ya ha aprendido.

d) Procesamiento rápido: Uno de los intereses más críticos en las tecnologías de procesamientos de datos ha sido siempre la velocidad de procesamiento. A partir de intentos de automatizar procesos que envuelven grandes volúmenes de información y de gran complejidad, ha surgido la necesidad cada vez más urgente, de desarrollar programas y equipos de alta eficiencia en el procesamiento de datos.

Tradicionalmente, las investigaciones en cuanto a eficiencia computacional se han dirigido primariamente al desarrollo de programas seriales y equipos que los hagan funcionar en forma rápida, por un lado mediante el aumento de eficiencia de los programas, a través del diseño de procesadores con arquitecturas mejoradas; y por otro, mejorando la capacidad y manejo de la memoria computacional.

Las redes neuronales han apuntado a paralelizar el procesamiento de datos mediante la división de tareas computacionales en pequeñas sub tareas, las cuales son repartidas sobre varios procesadores que operan simultáneamente.

e) Compatibilidad con las tecnologías existentes: Al igual como cualquier nueva idea o tecnología, un nuevo método de procesamiento como las redes neuronales tendría muy poca oportunidad de éxito y aceptación si fuera incompatible y difícil de integrar con otras tecnologías establecidas. Las redes neuronales son prácticas en este aspecto ya que pueden ser fácilmente implementadas utilizando técnicas existentes, y fácilmente integradas a casi cualquier tipo de sistema.¹⁹

8. Desventajas de las Redes Neuronales

Las Redes Neuronales Artificiales son modelos que ofrecen numerosas ventajas, pero también presentan algunos inconvenientes:

a) Las R.N.A. constituyen un método de resolución de problemas demasiado creativo, es decir, que dada las especificaciones de un problema, se desconoce la topología con la que se va a solucionar del modo más eficiente. Para ello hay que utilizar el ya conocido método de prueba y error.

b) Una vez entrenada una red neuronal, se hace difícil interpretar su funcionamiento. Aún más no es fácil asegurar con que grado de acierto responderá ante casos nunca vistos (generalización).

c) Es difícil averiguar por qué una red neuronal no es capaz de ajustar los datos que le son proporcionados. Esto implica que dado un problema, puede resultar dificultoso saber por qué no funciona correctamente, ya que existen una multitud de factores que pueden influir en su rendimiento.

d) Los modelos neuronales necesitan una herramienta de procesamiento poderosa.

¹⁹ [Jiménez 96c]

Esto se manifiesta principalmente en el proceso de aprendizaje, pero este inconveniente se contrarresta con la facilidad de implementación en dispositivos de *hardware* específicos.

IV. MODELOS DE PREDICCIÓN DE REDES NEURONALES

La necesidad de contar con técnicas que permitan predecir adecuadamente la evolución futura del valor del oro y que ayuden a las personas o inversionistas a tomar una mejor decisión respecto a la compra y venta de oro ha dado origen a diferentes técnicas de análisis. En este estudio podemos comprobar como una de las técnicas más importantes a nivel de predicción son las Redes Neuronales.

Cabe señalar que el principal objetivo de todos estos análisis es el de ser instrumentos de apoyo para el inversionista, ya que le entregan una mejor orientación, ayudándole a una buena toma de decisiones, determinando el momento óptimo para comprar o vender oro.

Hemos decidido predecir la dirección de la fluctuación del oro, antes que el nivel de precios ya que como han demostrado Leung, Daouk & Chen y que posteriormente fueron apoyados por O'Connor, Remous & Griggs (1997), la capacidad predictiva de los modelos de clasificación es mayor a la de los modelos con estimación de nivel, en términos de su tasa de acierto y son capaces de generar beneficios más altos. Cabe destacar que Hodgson & Nicholls (1991) sugieren evaluar la significancia económica de predecir la dirección de los cambios en los precios de los activos y no su nivel.²⁰

Las redes neuronales pueden entenderse como modelos multiecuacionales (o

²⁰ [Parisi 02d]

multi-etapas), simultáneas y/o secuenciales, en los que el output de unas constituye el input de otras. 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 imita el trabajo del cerebro humano.²¹

Las redes neuronales, nos ayudan a predecir el comportamiento de las personas a partir de datos que reflejan cómo se han comportado anteriormente. Las personas somos capaces de incorporar información en nuestro cerebro, y muchas veces lo hacemos de diversas fuentes y poseemos la capacidad de integrar toda esta información para luego aprender de ésta, encontrar patrones y extraer conclusiones que nos van a llevar a tomar decisiones en nuestra vida. Es así como las R.N.A. tratan de imitar todos estos “pasos” en el procesamiento de información que realizamos al tomar decisiones. Primero, se le entregan inputs al algoritmo, que va a ser la información con la cual se va trabajar. Ésta es procesada con el propósito que el programa pueda obtener algún grado de aprendizaje y después lo aplique en un conjunto de datos que no ha estudiado, para así poder obtener predicciones.

Para lograr obtener estos resultados vamos a utilizar dos técnicas de funcionamiento: recursivo y rolling aplicado en las redes neuronales. El funcionamiento rolling le otorga una mayor importancia a los datos que se tienen más recientemente y no toma en cuenta los datos más antiguos. El funcionamiento recursivo agrega nueva información, pero a la vez mantiene la que se tiene anteriormente.

Lo que se busca en este trabajo es, identificar el tamaño muestral óptimo y determinar cual de las técnica, si rolling o recursivo, posee la mayor capacidad predictiva, para luego medir la robustez de estos resultados utilizando bootstrap.

²¹ [Parisi 02e]

PARTE I: MODELOS ROLLING V/S RECURSIVOS

En esta primera parte se evaluará la capacidad de los modelos de redes neuronales y se comparará el porcentaje de predicción de signos de las dos técnicas dado distintos tamaños maestres. Para en una segunda etapa medir la robustez de estos resultados aplicando la técnica bootstrap a un modelo escogido, por su porcentaje de predicción de signo.

A. METODOLOGIA

1. Variables de entrada

Las variables escogidas en este estudio son las siguientes, las cuales fueron escogidas por su facilidad de uso y su correlación con la variable que se quiere predecir.

1.1 Diferencias Rezagadas del Oro

Se utilizan las diferencias rezagadas del oro, tanto el primer rezago como el segundo (δG_{t-1} , δG_{t-2}), por la correlación y la información que se puede obtener para el futuro con

los datos disponibles en el mercado los cuales reflejan las expectativas de los participantes del mercado.

1.2 Dow Jones Industry

Los promedios del Dow Jones tienen su origen en la iniciativa de varios editores financieros: Charles Henry Dow (1851-1902) y de Edward D. Jones (1856-1920). Ambos periodistas, quienes junto con Charles Bergstresser, fundaron en 1882 Dow Jones & Company.

Estos promedios son revisados por editores del diario Wall Street Journal y sus componentes varían cuando hay una adquisición de por medio o cuando hay cambios drásticos en el desempeño corporativo de una empresa. Si ocurre algún evento que amerita el reemplazo de un componente, el índice al que este pertenece es revisado por entero. Por esta razón, múltiples cambios son realizados simultáneamente.

Aunque no hay regulaciones explícitas para la selección de los componentes de cada promedio, los títulos son añadidos únicamente si tienen una excelente reputación, demuestran crecimiento sustancial, son parte del interés de varios inversores y representan de forma adecuada el sector al cual pertenecen.

El Promedio Industrial Dow Jones (DJIA), a diferencia del Promedio de Transportes Dow Jones (DJTA) y el Promedio de Utilidades Dow Jones (DJUA), no sólo limita los títulos tradicionalmente definidos como industriales. Por el contrario, sus componentes pertenecen a todo tipo de industrias: servicios financieros, tecnología, minoristas, entretenimiento y bienes del consumidor.

El DJIA, es sin duda el más seguido y reconocido índice de acciones. Lo conforma un diversificado portafolio con los títulos de 30 empresas de valor fijo alto (*blue chips stocks*), es por esto que se lo considera como un medidor del desempeño del mercado en general.

Los promedios del Dow tienen la particularidad de que se calculan sobre la base del precio y no de la capitalización. Por ende, el peso de los componentes varía de acuerdo al precio de las acciones, a diferencia de otros índices que son afectados no sólo por el precio sino también por el número de acciones en circulación.

Cuando los promedios fueron creados, su valor era calculado simplemente por la suma del precio de todos los componentes de cada índice y dividiéndolo por el número de componentes del mismo. Luego, se inició la práctica de utilizar un divisor para suavizar los efectos de la división de acciones (*stock split*) y otras estrategias corporativas.²²

2. Modelo

Se usaron 537 observaciones de precio de oro correspondientes a cierres diarios para el período comprendido entre el 5 de Enero de 2001 y el 28 de Febrero de 2003²³. El modelo formulado considera una variable de salida dada por la primera diferencia del

²² [www.terra.com]

precio del oro correspondiente al período “t” (δG_t), mientras que las variables de entrada se refieren a las primeras diferencias, rezagadas en uno y dos períodos de los precios del oro (δG_{t-1} , δG_{t-2}), y al índice bursátil Dow Jones Industry y a su primer rezago (δDJI_t y δDJI_{t-1}). La forma funcional del modelo se expresa en la ecuación (1):

$$\delta G = g(\delta G_{t-1}, \delta G_{t-2}, \delta DJI_t, \delta DJI_{t-1}) \quad (1)$$

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. Además, debido a que la información de hoy está desfasada con respecto a la proyección, se utilizan modelos dinámicos con rezago ($t-1$, $t-2$) en las variables independientes.

El modelo requiere para su funcionamiento que la base de datos se divida en a lo menos dos conjuntos diferentes: (a) conjunto de entrenamiento o muestral (conjunto de datos empleados para que la red aprenda el problema); (b) conjunto de producción o extramuestral (datos no incorporados en los conjuntos anteriormente mencionados, los cuales fueron usados para probar la capacidad de predicción de la red ante datos que nunca ha visto). Los datos del conjunto de entrenamiento se eligen de manera conveniente y dinámica, y su tamaño es determinado para cada iteración realizada por el programa, tanto para el caso recursivo como rolling. Por su parte, el conjunto extramuestral se determina para cada iteración como la diferencia entre el tamaño del conjunto de entrenamiento y el tamaño total de la muestra y abarca el mismo período de tiempo. Cabe mencionar que otros autores dividen el conjunto total de datos en dos segmentos, donde el primero es utilizado para entrenar la red mientras que el segundo es usado para determinar y validar la arquitectura y la especificación del modelo (Chen & Leung, 1998; Leung, Daouk & Chen, 2000).

B. Red Neuronal con funcionamiento recursivo y rolling

El análisis se centró en el empleo de una red neuronal multicapas con aprendizaje supervisado, específicamente la red Ward de tres capas. Se utilizó éste modelo de redes ya que actúan como funciones mapeadoras universales, desempeñándose muy bien con las series de tiempo. 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 & Skapura, 1993), se presenta en el Anexo N°2.

²³ Una tabla con las características estadísticas de las series utilizadas pueden encontrarse en el Anexo N°1.

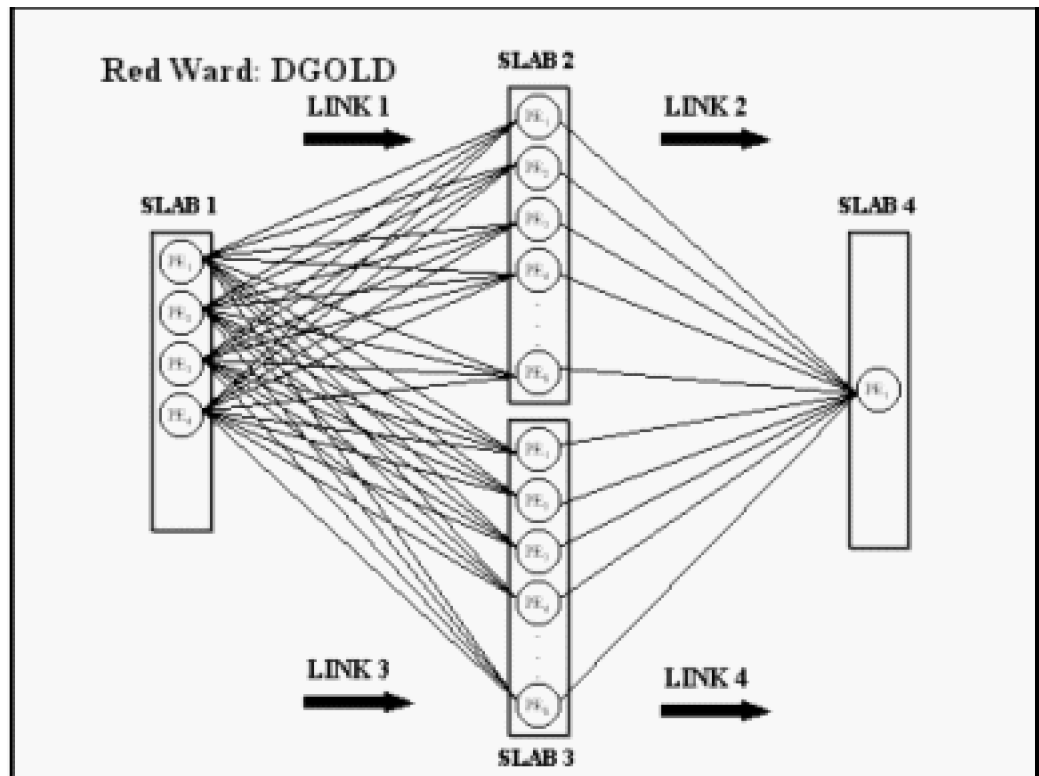


Figura 22: Red Ward

La red Ward cuenta con una capa de entrada, una capa oculta y otra de salida. En términos generales, la capa de entrada (Slab 1) posee una neurona por cada i -ésima variable de entrada, la capa oculta posee " j " neuronas (las cuales se dividen en partes iguales entre los Slab 2 y 3), y la capa de salida (Slab 4) posee una neurona. De ésta forma, la salida neta de la capa oculta viene dada por la ecuación (2):

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

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

$$\begin{aligned}
 Neta_{pj}^k &= \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1i} \\ w_{21} & w_{22} & \dots & w_{2i} \\ \vdots & \vdots & \dots & \vdots \\ w_{\left[\frac{j}{2}\right]_1} & w_{\left[\frac{j}{2}\right]_2} & \dots & w_{\left[\frac{j}{2}\right]_i} \end{bmatrix}_{\left(\left[\frac{j}{2}\right] \times i\right)}^k \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_i \end{bmatrix}_{(i \times 1)} + \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{\left[\frac{j}{2}\right]} \end{bmatrix}_{\left(\left[\frac{j}{2}\right] \times 1\right)}^k \\
 &= \begin{bmatrix} Neta_1 \\ Neta_2 \\ \vdots \\ Neta_{\left[\frac{j}{2}\right]} \end{bmatrix}_{\left(\left[\frac{j}{2}\right] \times 1\right)}^k \quad (3)
 \end{aligned}$$

La salida neta de la capa de salida queda expresada de acuerdo a la ecuación (4), para los Slabs 2 y 3:

$$o_{pk} = f_k^o(Neta_{pk}^o) \quad (4)$$

Donde:

$$\begin{aligned}
 &(5) \\
 Neta_{pk}^o &= \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1\left[\frac{j}{2}\right]} \\ \vdots & \vdots & \dots & \vdots \\ w_{\left[\frac{j}{2}\right]_1} & w_{\left[\frac{j}{2}\right]_2} & \dots & w_{\left[\frac{j}{2}\right]_i} \end{bmatrix}_{\left(1 \times \left[\frac{j}{2}\right]\right)}^o \times \begin{bmatrix} i_1 \\ i_2 \\ \vdots \\ i_{\left[\frac{j}{2}\right]} \end{bmatrix}_{\left(\left[\frac{j}{2}\right] \times 1\right)}^k + [\theta_1]_{(1 \times 1)}^p = [Neta_1]_{(1 \times 1)}^o
 \end{aligned}$$

Las características de la arquitectura usada para proyectar, junto con las funciones de activación, se presentan en el cuadro N°1.

Cuadro N°1: Funciones de activación y configuración de la Red Ward utilizada

	Red Ward			
	Slab 1	Slab 2	Slab 3	Slab 4
N° Pes	4	6	6	1
F. Activación	Tangente	Gausiana	Complemento Gausiana	Tangente

Como se señaló anteriormente, la red Ward fue aplicada empleando un funcionamiento recursivo primero y luego rolling. Así, se pretende reflejar que los precios responden a la formación de expectativas de los agentes del mercado, y éstas expectativas cambian con el transcurso del tiempo ante el surgimiento de nuevos eventos, cuestión que no es tomada en consideración por el funcionamiento estándar de la red neuronal. Por lo anterior, se planteó la necesidad de recalculer los pesos de la red Ward período a período, mediante un procedimiento iterativo a través de una técnica recursiva y rolling, con el objeto de ajustar el modelo al cambio de expectativas las cuales se explican a continuación.

1. Funcionamiento Recursivo

Esta metodología ha sido empleada anteriormente para medir el desempeño de modelos de redes neuronales que buscan predecir períodos de recesión en los Estados Unidos (Qi, 2001; Estrella & Mishkin, 1998) y para proyectar el signo de la variación de índices bursátiles internacionales (Parisi, Parisi & Guerrero, 2003). El funcionamiento recursivo consiste en agregar, mediante un algoritmo iterativo, nueva información a la ya estudiada por la red. Desde el punto de vista financiero, esto nos permite reflejar que es necesario que los agentes ajusten sus expectativas frente a la nueva información obtenida, pero sin dejar de considerar la totalidad de la información pasada, pues si razonan utilizando toda la información disponible sus predicciones serán más completas y por lo tanto, probablemente más acertadas. De esta forma, si consideramos que inicialmente 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 sale del conjunto extramuestral (quedando con “ $m-1$ ” datos) y pasa a formar parte del conjunto de entrenamiento, por lo que la muestra de “ n ” datos que contiene los valores de entrada se incrementa a “ $n+1$ ”. Luego se realizó una nueva iteración, lo que implicó reconstruir los pesos del 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, y , respectivamente. Este proceso se repitió hasta que en el conjunto extramuestral quedó sólo un dato, por lo que la red recalculó los pesos y 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.

2. Funcionamiento Rolling

El funcionamiento rolling pretende capturar el proceso de adaptación de expectativas, de

manera tal que se le otorgue una mayor importancia a aquella información recientemente agregada y descartando la más antigua, simulando el hecho de que los agentes ajustan sus expectativas dando una mayor ponderación a la información más reciente. Específicamente, éste mantiene constante el tamaño del conjunto muestral “ n ”, pero varía desde el punto de vista de su composición, pues para cada iteración éste se va desplazando a través del conjunto total de datos, de manera que incluye la próxima observación pero a la vez elimina la más antigua “ $-1+n+1=n$ ”. Como en el caso recursivo, el proceso se repite “ $m-1$ ” veces, permitiéndole a la red recalcular sus pesos de acuerdo a lo aprendido.

C. Evaluación de la capacidad predictiva

Cuadro N°2: Tamaño de los conjuntos muestrales y extramuestrales

Tamaño Conjunto Muestral	Tamaño Conjunto Extramuestral
30 datos	507 datos
35 datos*	502 datos
38 datos*	499 datos
40 datos	497 datos
42 datos*	495 datos
45 datos*	493 datos
48 datos*	490 datos
50 datos	487 datos
52 datos*	485 datos
55 datos*	482 datos
58 datos*	479 datos
*Tamaños maestres solamente utilizados para el funcionamiento Rolling	

La capacidad de los modelos de redes neuronales para predecir el signo de las variaciones del precio del oro de acuerdo al tamaño de muestra y la técnica recursiva, fue contrastada con la capacidad predictiva de un tamaño de muestra similar para la técnica rolling. De ésta forma, se pretende encontrar si existen combinaciones óptimas de técnica y tamaño muestral que permitan obtener mejores rendimientos en términos de aciertos en el signo de la predicción.

1. Análisis De Resultados

Luego de probar las distintas combinaciones entre el tamaño muestral y la técnica utilizada, los mejores resultados obtenidos fueron los siguientes: para un mismo tipo de red, la técnica rolling alcanzó un rendimiento de 73,05% con un tamaño muestral de 52 datos acertando el signo en 356 períodos diarios, mientras que la técnica recursiva

registró una capacidad predictiva de 64,39% para un tamaño muestral de 30 datos, con 319 aciertos. El cuadro N°3 presenta un resumen de los resultados para el total de las combinaciones estudiadas. Así, las redes Wards rolling superaron en capacidad predictiva a las redes con funcionamiento recursivo. Se destaca que de acuerdo al test DA, todas las redes estudiadas registraron una capacidad predictiva significativa en términos estadísticos.

Los resultados anteriores se ajustan a lo esperado, debido a que el funcionamiento recursivo y rolling actualizan la información que el modelo emplea para realizar las proyecciones, y lo hace recalculando los pesos período a período. Sin embargo, el funcionamiento rolling supera el funcionamiento recursivo, lo cual puede interpretarse como que efectivamente los agentes estudian la información pasada, pero sólo relevante a un período de tiempo reducido. Sin perjuicio de lo anterior, los resultados obtenidos en esta investigación apoyan el uso de redes neuronales con funcionamiento rolling y en menor medida el funcionamiento recursivo, lo cual supone que en el caso del precio del oro, la variabilidad de los pesos a través del tiempo es significativa y, por lo tanto, el conjunto muestral no contiene toda la información necesaria para entrenar la red, presentando evidencia en el sentido de que los precios responden a la formación de expectativas, y éstas expectativas cambian con el transcurso del tiempo ante el surgimiento de nuevos eventos.

2. Conclusiones parte I

De las diversas arquitecturas de redes neuronales utilizadas en este estudio, las redes Wards rolling superaron en capacidad predictiva a las redes con funcionamiento recursivo. Estos resultados respaldan el uso de las redes neuronales con funcionamiento dinámico para predecir el signo de las variaciones del precio del oro, por lo que sería necesario recalcular los pesos de la red período a período, a través de un proceso rolling, ya que el conjunto de entrenamiento no contendría información acerca de todos los posibles escenarios con los que podría encontrarse la red en el futuro. Cabe señalar que Parisi, Parisi & Guerrero (2003) encontraron que, para el caso de índices bursátiles internacionales, la capacidad predictiva de los modelos de redes neuronales con funcionamiento recursivo varía a través del tiempo lo que es validado aquí.

De acuerdo al test DA, la red Ward rolling para un tamaño muestral de 52 datos registró una capacidad predictiva significativa en términos estadísticos, alcanzando el mejor desempeño en términos de predicción de signo. De esta forma, la capacidad predictiva de los modelos de redes neuronales dinámicas resulta relevante, lo que los sitúa como una alternativa al análisis técnico y a las técnicas convencionales de proyección de series de tiempo, como los modelos ARIMA.

PARTE II. MEDICIÓN DE LA ROVUSTED UTILIZANDO BOOTSTRAP

En esta segunda parte se evaluará la robustez de los resultados obtenidos en la primera parte utilizando bootstrap. La cual es una técnica estadística para poder realizar inferencia, obtener la distribución de los resultados, establecer intervalos de confianza, etc. Esta técnica consiste en generar nuevas series, combinando aleatoriamente las series ya existente.

A. METODOLOGÍA

1. Modelo

Se escogió la técnica del rolling para medir su robustez, dado el mejor desempeño logrado por esta técnica sobre un funcionamiento recursivo, el rolling escogido fue el de tamaño muestral de 30 datos, el cual deja como marco extramuestral 507 datos, el cual obtuvo una capacidad de predicción de signo de 72.19%.

2. Bootstrap

Una serie bootstrap, consiste en crear series ficticia que se obtiene a partir de los datos originales, construyendo con esto datos distintas series aleatorias, las cuales se fabrican con combinaciones de bloques de la serie original, al ser datos diarios, en este caso los bloques fueron de 5 datos, para conservar una suerte de tendencia semanal, la cual no implica que el bloque partiera un lunes y terminara un viernes, el inicio del bloque se tomaba aleatoriamente de dentro de las 537 observaciones, con un numero aleatorio “n” entre 1 y 532, en ese punto comenzaba un bloque y terminaba cinco datos después., luego se obtenía otro “n” aleatoriamente y se unía al bloque ya creado hasta obtener una nueva serie ficticia de 537.

Se crearon 53 nuevas series a las cuales se les corrió el modelo de red Ward de tres capas, con funcionamiento rolling y con un tamaño muestral de 30 datos.

B. ANÁLISIS DE LOS RESULTADOS

los resultados obtenidos son concluyentes y demuestran que la aplicación de las redes neuronales y los altos porcentajes de predicción de signos que obtiene no son una casualidad temporal, por lo tanto las podemos inferir dado los resultados que se presentan a continuación que las redes obtiene un alto desempeño sin importar el escenario económico sobre el cual se busque obtener una buena predicción. El cuadro N°4 presenta un resumen de los resultados para el total de las series bootstrap estudiadas. Se destaca que de acuerdo al test DA, todas las redes estudiadas registraron una capacidad predictiva significativa en términos estadísticos.

Cuadro N° 4: Resumen de los resultados de las series bootstrap

Serie	Predicción	Rentabilidad	Buy & Hold	DA	Predictivo*
serie 1	72.84%	586.50%	166.00%	7.92	Sí
serie 2	77.87%	1371.28%	108.95%	9.39	Sí
serie 3	78.87%	1482.94%	63.05%	9.96	Sí
serie 4	74.25%	895.44%	76.65%	7.98	Sí
serie 5	78.87%	935.81%	148.20%	9.95	Sí
serie 6	78.47%	2007.29%	80.75%	9.43	Sí
serie 7	78.27%	752.14%	138.70%	8.89	Sí
serie 8	81.09%	2119.24%	44.65%	10.10	Sí
serie 9	78.07%	1351.21%	101.05%	9.64	Sí
serie 10	77.06%	1322.37%	73.50%	9.57	Sí
serie 11	79.15%	889.46%	101.70%	9.99	Sí
serie 12	78.64%	853.42%	116.60%	9.26	Sí
serie 13	77.39%	1159.34%	79.75%	7.19	Sí
serie 14	80.15%	910.88%	130.05%	9.68	Sí
serie 15	77.64%	785.92%	123.10%	7.89	Sí
serie 16	72.11%	858.24%	80.10%	7.58	Sí
serie 17	81.16%	797.97%	158.65%	9.07	Sí
serie 18	77.39%	1415.78%	47.15%	8.54	Sí
serie 19	80.40%	858.57%	173.90%	8.73	Sí
serie 20	68.84%	637.30%	76.10%	4.88	Sí
serie 21	76.66%	825.08%	93.05%	8.76	Sí
serie 22	71.83%	1196.53%	110.30%	6.15	Sí
serie 23	75.05%	977.77%	67.40%	8.65	Sí
serie 24	75.05%	837.02%	151.00%	8.86	Sí
serie 25	73.64%	2873.56%	35.20%	8.24	Sí
serie 26	66.00%	772.08%	20.60%	4.28	Sí
serie 27	75.05%	1977.06%	28.10%	8.69	Sí
serie 28	76.26%	986.42%	98.95%	9.51	Sí
serie 29	73.84%	714.85%	127.80%	8.00	Sí
serie 30	76.13%	856.76%	123.10%	9.37	Sí
serie 31	77.12%	714.20%	158.45%	9.56	Sí
serie 32	75.94%	2176.49%	35.85%	8.49	Sí
serie 33	78.11%	842.73%	201.00%	9.95	Sí
serie 34	79.09%	1057.97%	120.40%	9.61	Sí
serie 35	80.08%	2301.83%	69.35%	9.61	Sí
serie 36	77.51%	1897.91%	61.05%	9.49	Sí
serie 37	76.92%	1044.89%	99.35%	9.17	Sí
serie 38	80.08%	2249.02%	59.60%	11.22	Sí
serie 39	77.12%	927.54%	152.65%	8.51	Sí
serie 40	78.30%	1950.35%	57.55%	10.09	Sí
serie 41	77.12%	1675.26%	68.05%	9.31	Sí
serie 42	76.13%	856.76%	123.10%	9.37	Sí
serie 43	77.12%	714.20%	158.45%	9.56	Sí
serie 44	75.94%	2176.49%	35.85%	8.49	Sí

“REDES NEURONALES APLICADAS A LA PREDICIÓN DEL PRECIO DEL ORO Y MEDICIÓN DE LA ROBUSTEZ DE LOS RESULTADOS UTILIZANDO BOOTSTRAP”

serie 45	78.11%	842.73%	201.00%	9.95	Sí
serie 46	79.09%	1057.97%	120.40%	9.61	Sí
serie 47	80.08%	2301.83%	69.35%	9.61	Sí
serie 48	77.51%	1897.91%	61.05%	9.49	Sí
serie 49	76.92%	1044.89%	99.35%	9.17	Sí
serie 50	80.08%	2249.02%	59.60%	11.22	Sí
serie 51	77.12%	927.54%	152.65%	8.51	Sí
serie 52	78.30%	1950.35%	57.55%	10.09	Sí
serie 53	77.12%	1675.26%	68.05%	9.31	Sí
Promedio	76.96%	1293.23%	98.75%	-	-
Mínimo	66.00%	586.50%	20.60%	-	-
Máximo	81.16%	2873.56%	201.00%	-	-
La capacidad predictiva es determinada con un nivel de confianza del 95%. Valor Z crítico = 1,96					

En las figuras 23 y 24 se presentan los histogramas de la capacidad de predicción de signos y las rentabilidades obtenidas por la técnica respectivamente, además de presentarse los estadísticos necesarios para medir normalidad.

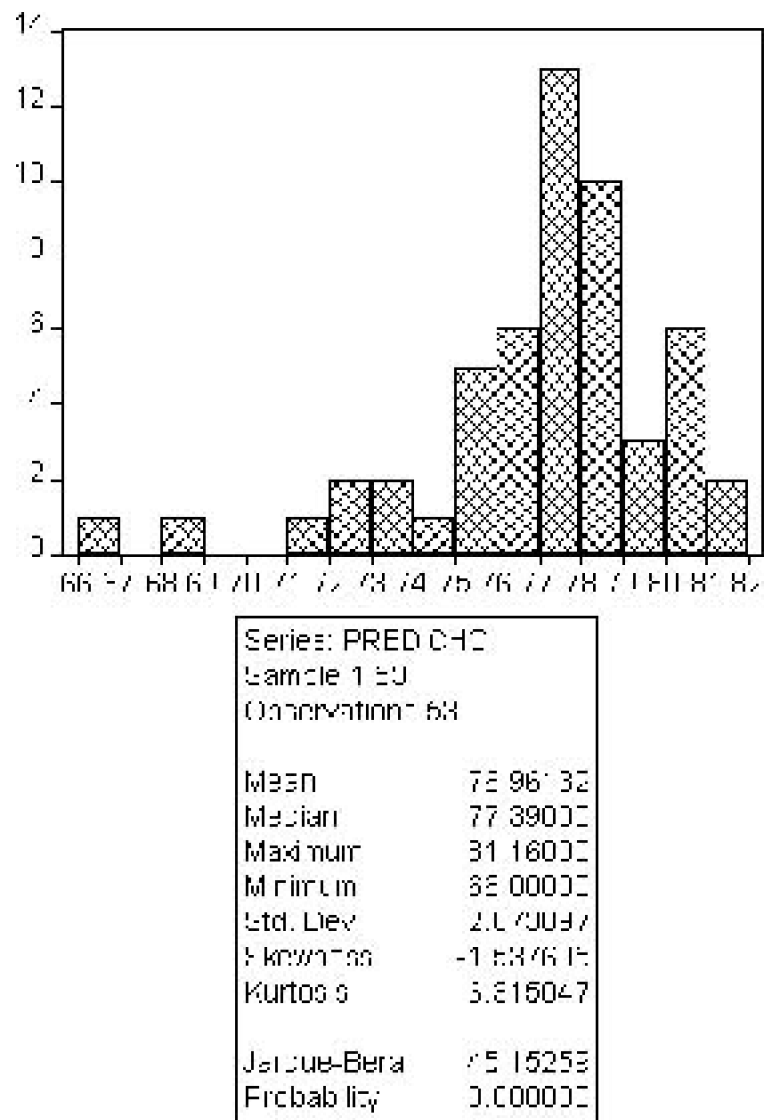


FIGURA 23: Histograma Porcentaje Predicción de Signos Series Bootstrap

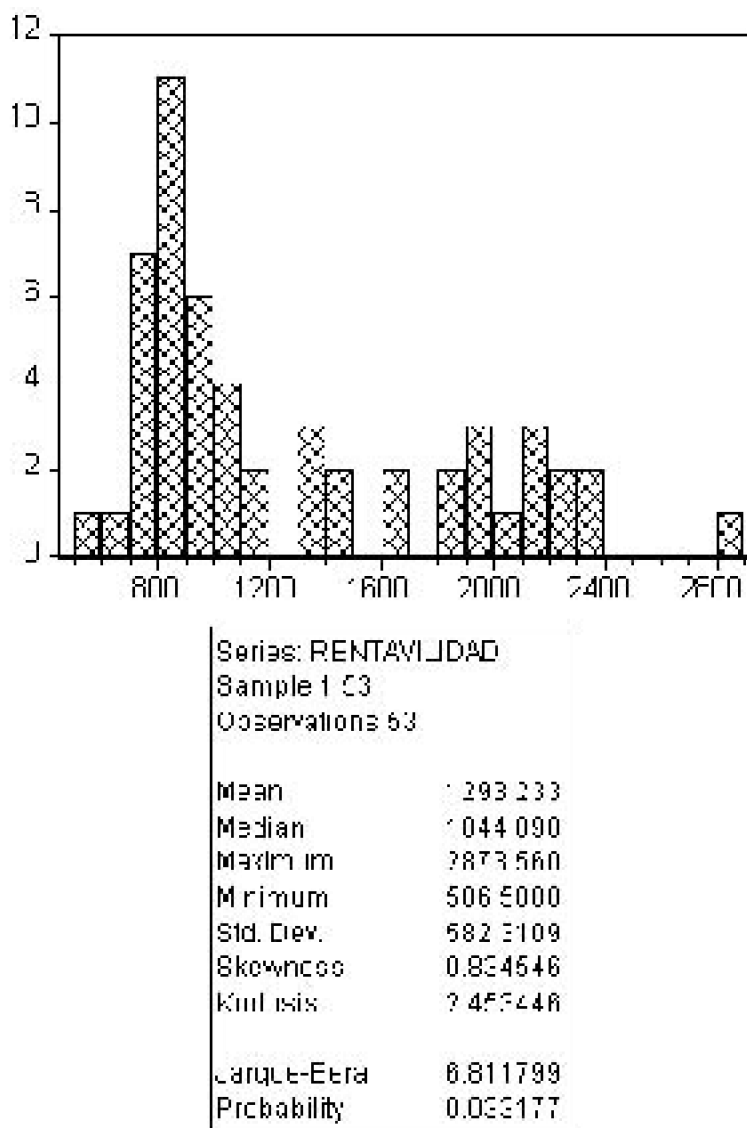


FIGURA 24: Histograma de la Rentabilidad de las Series Bootstrap

1. Conclusiones Parte II

Podemos concluir que el modelo es robusto, dado que todas las series obtienen retornos sobre normales y el porcentaje de predicción de signos, de cada serie, se encuentra dentro del rango de 66% y 81,16%, predicción que siempre es significativa, según el test DA. Lo que avala el uso de las redes neuronales para la predicción de variaciones de precios diarios del oro, bajo cualquier escenario económico.

V. CONCLUSIONES FINALES

El modelar mercados financieros a través de Redes Neuronales Dinámicas, requiere poseer una adecuada información de lo que se desea medir, como también de las variables que se van a utilizar, el número de datos que va a contener el tamaño del Rolling, la arquitectura, entre otros. De este estudio se desprende que:

La técnica de reconstruir los modelos con un funcionamiento rolling, da mejores resultados que la reconstrucción de modelos con un proceso recursivo, lo que validaría la hipótesis de una memoria de corto plazo, en lo mercados financieros.

Por último, dado los resultados obtenidos por las redes neuronales dinámicas en las series bootstrap, se puede concluir que esta herramienta se puede utilizar en cualquier escenario económico, lo que valida su uso para la predicción del oro, ya que los modelos de redes neuronales rolling podrían mejorar la capacidad de predicción de los movimientos del precio del oro, lo que permitiría elevar la rentabilidad y reducir el riesgo asociado a las transacciones de dicho metal.

VI. BIBLIOGRAFIA

- 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.
- Chen, A. & Leung, M. (1998). Dynamic foreign currency trading guided by adaptive forecasting. *Review of Pacific Basin Financial Markets and Policies* 1, p.p. 383-418.
- Estrella, A., & Mishkin, F. S. (1998). Predicting US recessions: Financial variables as leading indicators. *The Review of Economics and Statistics* 80(1), p.p. 45-61.
- Freeman, J. & 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.
- Herbrich, R., Keilbach, M., Graepel, T., Bollmann-Sdorra, P. & Obermayer K. (2000). "Neural networks in economics: Background, applications and new developments". In T. Brenner, editor, *Advances in Computational Economics: Computational Techniques for Modelling Learning in Economics*, volume 11, pages 169-196. Kluwer Academics.
- Hodgson, A. & Nicholls, D. (1991). The impact of index futures markets on Australian share market volatility. *Journal of Business Finance and Accounting* 18, p.p. 267-280.

- Kanas, Angelos (2001). “Neural networks linear forecasts for stocks returns”. *International Journal of Finance and Economics*, 6, p.p. 245-254.
- Leung Mark T., Daouk Hazem, Chen An-Sing (2000). “Forecasting stock indices: a comparison of classification and level estimation models”. *International Journal of Forecasting* (16)2, pp.173-190.
- MacCann Peter J. , Kalmann Barry L. (1994).“A Neural Network Model for the Gold Market” *Journal of Forecasting* 16, p.p. 165-176.
- O’Connor, M., Remus, W., & Griggs, K. (1997). Going up-going down: How good are people at forecasting trends and changes in trends? *Journal of Forecasting* 16, p.p. 165-176.
- Odom M. D. & Sharda R. (1990). A neural network model for bankruptcy prediction. *Proceedings of the IEEE International Conference on Neural Networks*, San Diego II, p.p. 163-168.
- Parisi, A., Parisi, F. & Guerrero, J. L. (2003). “Modelos Predictivos de Redes Neuronales en Índices Bursátiles”, *El Trimestre Económico*, México. (Forthcoming).
- Pesaran, M.H. & Timmermann A. (1992). A simple nonparametric test of predictive performance. *Journal of Business and Economic Statistics* 10, pp. 461-465.
- 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. & 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.
- Tang, Z., de Almieda C.& Fishwick P. (1991). Time-series forecasting using neural networks vs. Box-Jenkins methodology. *Simulation*, 57(5), p.p. 303-310.
- Tsibouris, G. & 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.
- 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.
- Wilson, J. H. & Keating B. (1998). *Business Forecasting*. Third Edition, U.Irwin/McGraw-Hill, U.S.A.
- Wilson, R. L. & Sharda R. (1994). Bankruptcy prediction using neural networks. *Decision Support Systems*, 11(5), June, p.p. 545-557.
- 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.
- [Del Brío 97a] Del Brío, Bonifacio Martín & Sanz Molina, Alfredo, “Redes Neuronales y Sistemas Borrosos”, Madrid, RAMA, 1997, p. 11, 12.
- [Espinoza 00] Espinoza Medina, Hernán Paul, “Predicción de Variables Financieras Utilizando Redes Neuronales”, Memoria de Ingeniero Civil Electricista, Universidad

de Chile, 2000.

- [EspinozaMu 02a] Espinoza Muñoz, Christian, "Presentación Redes Neuronales: El Análisis Técnico en su expresión más avanzada", Universidad de Chile, 2002.
- [Jiménez 96a] Jiménez Carvacho, Carlos, "Predicción de Acciones Utilizando Redes Neuronales", Memoria de Ingeniero Civil Industrial, Universidad de Chile, 1996, p. 17.
- [Jiménez 96b] Jiménez Carvacho, Carlos, "Predicción de Acciones Utilizando Redes Neuronales", Memoria de Ingeniero Civil Industrial, Universidad de Chile, 1996, p. 40, 41.
- [Jiménez 96c] Jiménez Carvacho, Carlos, "Predicción de Acciones Utilizando Redes Neuronales", Memoria de Ingeniero Civil Industrial, Universidad de Chile, 1996, p. 18, 19, 20.
- [Larraín 90a] Larraín Sáenz, Roberto, "Análisis de las Variables Influyentes en la Predicción de Precios de Acciones", Memoria de Ingeniero Industrial, Universidad de Chile, 1990, p. 35.
- [Parisi 02a] Parisi F., Antonino, "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA", Estudios de Administración, vol. 9. N°1, Universidad de Chile, 2002, p. 77.
- [Parisi 02b] Parisi F., Antonino, "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA", Estudios de Administración, vol. 9. N°1, Universidad de Chile, 2002, p. 78.
- [Parisi 02c] Parisi F., Antonino, "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA", Estudios de Administración, vol. 9. N°1, Universidad de Chile, 2002, p. 79.
- [Parisi 02d] Parisi, Antonino, "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA", Estudios de Administración, vol. 9, N° 1, Universidad de Chile, 2002, p. 71.
- [Parisi 02e] Parisi, Antonino, "Evaluación de modelos de redes neuronales de predicción del signo de la variación del IPSA", Estudios de Administración, vol. 9, N° 1, Universidad de Chile, 2002, p. 76.
- [ParisiRoll 03a] Parisi, Antonino, "Modelos de redes neuronales Rolling: tamaño muestral óptimo", Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2003.
- [ParisiOro 03a] Parisi, Antonino, "Modelos de redes neuronales Rolling versus Recursivo: Precio del Oro", Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2003.
- [Pesaran 92] Pesaran, M. H. & Timmermann A., "A Simple Non Parametric Test of Predictive Performance", Journal of business and economic statistics, 1992.
- [Reyes 02a] Reyes, Daniel & Tapia, Víctor, "Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros", Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 7.
- [Reyes 02b] Reyes, Daniel & Tapia, Víctor, "Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros", Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 8, 9.

- [Reyes 02c] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 10.
- [Reyes 02d] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 11, 12.
- [Reyes 02e] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 13, 14.
- [Reyes 02f] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 15.
- [Reyes 02g] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 16.
- [Reyes 02h] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 17.
- [Reyes 02i] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 18.
- [Reyes 02j] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 32, 33.
- [Reyes 02k] Reyes, Daniel & Tapia, Víctor, “Evaluación de los distintos modelos de redes neuronales en la predicción de valores financieros”, Facultad de Ciencias Económicas y Administrativas, Universidad de Chile, 2002, p. 42.
- [Rochette 94] Rochette García, Alain Emile, “Utilización del análisis técnico en los mercados financieros. Aplicación a la tasa de interés de largo plazo”. Memoria de Ingeniero Civil Industrial, Universidad de Chile, 1994.
- [Sanhueza 98] Sanhueza, Gonzalo, “Balanza Comercial y Política Cambiaria”, Jornadas de la Salmonicultura, Instituto Tecnológico del Salmón S.A., Septiembre de 1998, p. 7.
- [Virreira 96a] Virreira Zijderveld, Roberto, “Aplicación de Redes Neuronales a la Predicción y Transacción en Mercados Cambiarios”, Memoria Ingeniería Industrial de la Universidad Católica, 1996, p. 15.
- [Virreira 96b] Virreira Zijderveld, Roberto, “Aplicación de Redes Neuronales a la Predicción y Transacción en Mercados Cambiarios”, Memoria Ingeniería Industrial de la Universidad Católica, 1996, p. 10.
- [Virreira 96c] Virreira Zijderveld, Roberto, “Aplicación de Redes Neuronales a la Predicción y Transacción en Mercados Cambiarios”, Memoria Ingeniería Industrial de la Universidad Católica, 1996, p. 21.
- [Virreira 96d] Virreira Zijderveld, Roberto, “Aplicación de Redes Neuronales a la

Predicción y Transacción en Mercados Cambiarios”, Memoria Ingeniería Industrial de la Universidad Católica, 1996, p. 16.

[Virreira 96e] Virreira Zijderveld, Roberto, “Aplicación de Redes Neuronales a la Predicción y Transacción en Mercados Cambiarios”, Memoria Ingeniería Industrial de la Universidad Católica, 1996, p. 11.

Anexos.

[Anexos, formato PDF \(201 KB\)](#)