

LAS REDES NEURONALES EN SU APLICACIÓN A LAS FINANZAS

*Ruiz Martínez, Ramón Jesús
Jiménez Caballero, José Luis*

Universidad de Sevilla

INDICE:

- 1. ¿Qué son las redes neuronales?**
 - 1.1. Introducción**
 - 1.2. Concepto y descripción**
 - 1.3. Funcionamiento de los sistemas neuronales artificiales**
- 2. ¿Cómo diseñar una aplicación?**
- 3. ¿Cuándo es útil la aplicación de RNA?**
- 4. Aplicaciones de las RNA**
- 5. Sistemas neuronales empleados en los mercados financieros de renta variable**
- 6. Aplicación práctica de las redes neuronales al mercado financiero español**

1. ¿QUÉ SON LAS REDES NEURONALES?

1.1.Introducción.

El ser humano en su intento de captar y medir la realidad circundante ha tratado de realizar una cuantificación de la misma, tradicionalmente, mediante la aplicación de representaciones matemáticas. De esta forma, la compleja realidad ha intentado ser introducida en la horma establecida por los modelos existentes, en ocasiones, con tal rigidez en sus planteamientos que no ha sido posible emularla con un mínimo de precisión. Lo único que se ha conseguido con ello es adaptarse a los fallos que presentan los modelos, cuando lo razonable debería ser apostar por el diseño de nuevos patrones más ajustados a nuestro objetivo final.

Ahora bien, teniendo en cuenta que los sistemas físicos han tenido un desarrollo, desde el punto de vista de su comprensión, bastante superior al logrado por otras disciplinas como las ciencias sociales o económicas; y que de la observación de un sistema social o económico podríamos colegir la misma estructura de comportamiento que la percibida en uno físico o biológico; es decir, altamente organizada, realimentada y no lineal [Forrester, 1998], la construcción de modelos que aprovechen las potencialidades y ventajas de los sistemas biológicos (p.e. el funcionamiento del cerebro humano) podría constituir un avance en la búsqueda de una mejor representación del entorno presente.

En este sentido, las redes neuronales artificiales (en adelante RNA) aparecen como una alternativa que trata de representar de manera adecuada una compleja realidad que presenta información masiva, redundante, imprecisa y distorsionada; unas veces, con mayor acierto que otros procedimientos, y otras, en las que habrá que guardar un mínimo de reserva ante sus resultados. Esta tecnología va a permitir la incorporación de *inteligencia* a un sistema de procesamiento y control, y contribuirá a la construcción de máquinas que emulen la capacidad humana de toma de decisiones en entornos imprecisos y con ruido. Ofrece, también, una nueva visión para el estudio de fenómenos cognitivos, pudiendo contribuir, de forma importante, a su comprensión [Martín del Brío,1997].

Nuestro propósito, por tanto, se dirige a intentar conocer el funcionamiento genérico de esta técnica, sus posibles aplicaciones en el área financiera y, más concretamente, en el ámbito de los mercados de valores de renta variable (predicción de cotizaciones, principalmente). Todo ello nos permitirá conocer la idoneidad y oportunidad de utilizar las RNA, en comparación a otros métodos conocidos.

1.2. Concepto y descripción.

No existe una definición universalmente aceptada de RNA; así, autores como Hetch-Nielsen (1988) entienden por red neuronal artificial el “*sistema de computación, formado de un gran número de elementos simples, altamente interconectados, y que procesan la información mediante respuestas dinámicas a unos estímulos externos*”. En

este mismo sentido, T. Kohonen (1988) entiende las RNA como *“redes en paralelo conectadas masivamente (normalmente adaptativas) de elementos simples y con organizaciones jerárquicas que son proyectadas para interaccionar con los objetivos del mundo real de la misma forma que lo hacen los sistemas nerviosos naturales”*. Cox (1993) la define como *“un sistema adaptativo que resuelve problemas mediante la imitación de la arquitectura de las redes neuronales naturales”*. Terminaremos, con una definición aportada por Sigüenza, López y Dorronsoro (1993) donde se pone de relieve que una red neuronal *“está formada por un conjunto de unidades que intercambian información cuyo resultado es un mensaje que envía a sus vecinas, pero que en conjunto es una computación bastante elaborada”*.

Se observa en estas definiciones que las RNA tienen como notas comunes, por una parte, la existencia de elementos simples que procesan información, y, por otra, que se produce una interconexión entre dichos elementos. Esta filosofía de proceso está inspirada en las ventajas que proporciona la estructura y funcionamiento del cerebro humano. Así, hasta hace poco tiempo, se hacía bastante difícil el planteamiento y solución de un problema complejo con la mera existencia de ordenadores, en los que la estructura hardware se constituía en torno a un único microprocesador complejo (CPU) que organizaba y procesaba, de forma secuencial, la información que le suministraba la memoria. La estructura hardware del cerebro humano se compone, sin embargo, de miles de millones de pequeños procesadores simples, llamados neuronas, de menor potencia que la CPU, que interactuando en paralelo logran solucionar problemas que no están al alcance de los ordenadores.

De este modo, considerando que hay aspectos donde el cerebro humano es capaz de realizar funciones con una eficacia y precisión superiores a cualquier computadora, la creación de una estructura de interconexión multineuronal podría ayudarnos a emular las habilidades del mismo. En este sentido, las redes neuronales artificiales son modelos simplificados que tratan de imitar la estructura y funcionamiento de las redes neuronales biológicas, y cuyos componentes fundamentales son las neuronas artificiales.

Siguiendo a Martín del Brío (1997), las tres características que las RNA van a tratar de *copiar* de los sistemas nerviosos biológicos consisten en el procesamiento en **paralelo**, la memoria **distribuida** y la **adaptabilidad**.

En cuanto al primer aspecto, la actuación de las neuronas en paralelo garantiza mayor rapidez en el proceso de información que la aportada por una operación secuencial, realizada instrucción a instrucción.

En segundo término, los sistemas neuronales poseen información que se distribuye ampliamente por toda la red. De este modo, cualquier deterioro neuronal afectaría mínimamente al sistema y, teniendo en cuenta que las redes nerviosas biológicas son redundantes, podríamos encontrar información acumulada en otras neuronas.

En último lugar, las RNA deben adaptarse al entorno. Para ello el aprendizaje a partir de ejemplos debe permitirles modificar el peso de las conexiones neuronales para lograr una generalización del problema a tratar.

Una vez fijados el concepto y algunas características fundamentales de las Redes Neuronales, vamos a realizar una descripción general de sus elementos constitutivos básicos y de su funcionamiento.

1.3. Funcionamiento de los sistemas neuronales artificiales

La estructura de una RNA está compuesta, en primer lugar, por un elemento esencial, la **neurona artificial**, que se configura como un elemento de cálculo básico que recibe información del exterior o de otras neuronas, y se transforma en una salida única. La neurona artificial (fundamentada en su homónima biológica) tiene una estructura de entradas, las **dendritas**, por las cuales accede información al **cuerpo celular** o soma, el cual procesa y responde al estímulo recibido, en función de la intensidad asignada al nexo de unión entre las entradas y la neurona¹. La respuesta de la neurona producirá una salida por el denominado **axón** (figura 1). Según el procesamiento que haga una neurona de la información recibida, podrá enviar una respuesta a la siguiente, o por el contrario no hacerlo. Ello dependerá del peso sináptico asignado a cada entrada, es decir, de la ponderación o importancia relativa asociados al estímulo recibido.

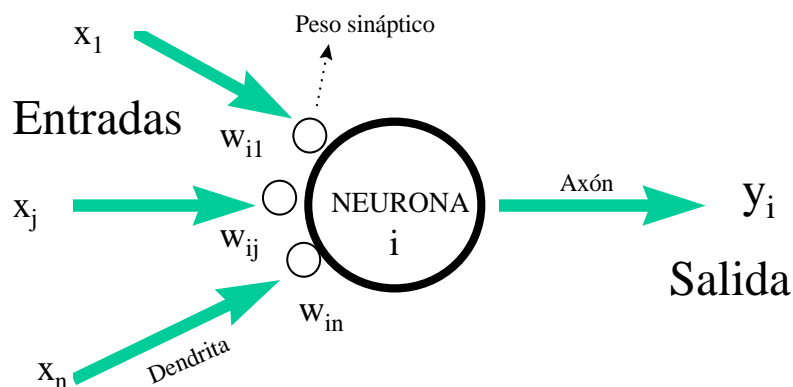


Figura 1: Neurona Artificial

Las neuronas se agrupan en estructuras unitarias denominadas **capas**, donde se produce una conexión de los niveles anteriores con los posteriores mediante la unión axón-dendrita. Así, la primera capa de la red, la de entrada, será diseñada en función de los datos externos ligados al problema, y dirigirá su salida hacia la siguiente; a continuación pueden aparecer capas intermedias u ocultas sin conexión con el entorno e interconectadas con las precedentes y siguientes, cuyo número dependerá de la dificultad asociada al problema planteado; finalmente existe una capa de salida cuya constitución y resultado debe responder al problema propuesto. De esta forma, el conjunto de capas e interconexiones se constituye en la arquitectura o topología de la **red neuronal**.

¹ Este nexo se denomina sinapsis, y a la graduación que se le asigna, peso sináptico.

Por otro lado, es necesario dotar a la red de una regla o **algoritmo de aprendizaje** que vaya ajustando, progresivamente, los resultados finales de la red a los deseados por el diseñador de la misma. El conjunto de entradas, salidas, red o topología neuronal y algoritmo de aprendizaje utilizados formará un **sistema neuronal** (véase figura 2).

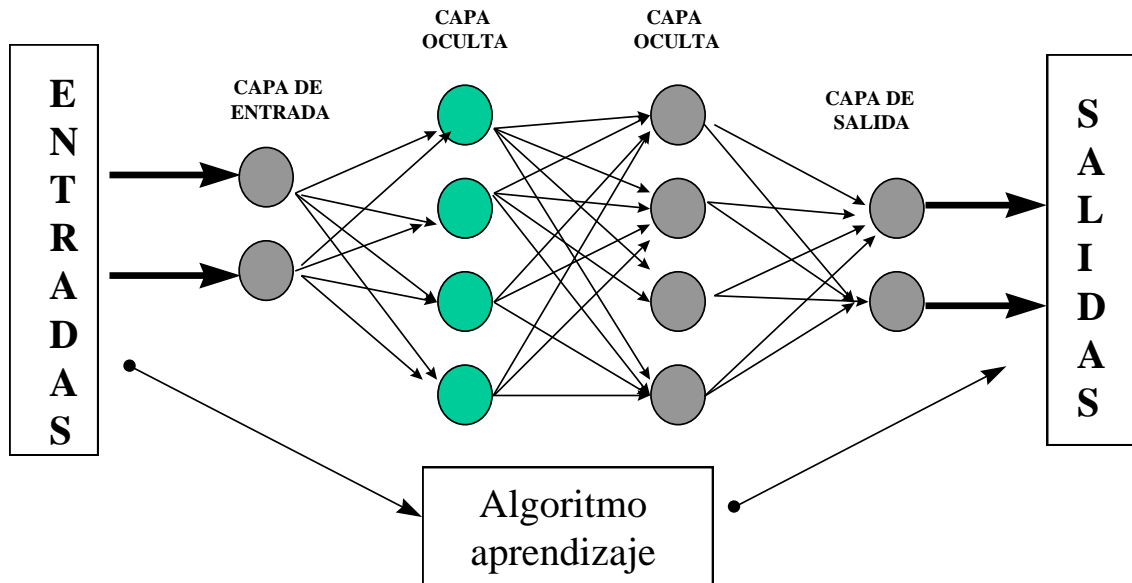


Figura 2: Estructura de un sistema neuronal artificial de cuatro capas

Pero entremos más detenidamente a observar el funcionamiento de un sistema neuronal.

La unidad elemental del sistema, la neurona, recibe unas entradas, x_j , sobre las que actúa una matriz de pesos iniciales aleatorios, w_{ij} , obteniéndose unos valores y_i , (ver figura 1) como resultado de aplicar una **función de entrada**. Su forma más habitual es la lineal², que se basa en la suma ponderada de las entradas por los pesos sinápticos:

$$y_j = \sum_j w_{ij} x_j$$

Este resultado debe ser filtrado a través de una **función de activación o transferencia** que convierte la resultante en unos valores que están dentro de unos límites de salida, y que servirá como dato de entrada a la siguiente neurona o capa. Con ello, se logra modificar los niveles de salida -que podrían ser muy amplios, e incluso infinitos- para que alcancen valores razonables y finitos (v.g. entre cero y uno)³. Asimismo, la no linealidad que puede mostrar la estructura de esta función reporta

² Se pueden encontrar otros tipos como la función radial básica, función sigma pi, distancia L1, L2, etc.

³ Cuando la función alcance el valor cero no se transmitirá ninguna señal a la siguiente neurona, que sólo será activada en caso de que el valor se haga uno.

ventajas a las redes neuronales sobre los sistemas de regresión clásicos, enriqueciendo su capacidad de representación. Las funciones más habitualmente utilizadas suelen ser la de identidad, escalón o limitador, umbral o lineal a tramos y sigmoide. Otras funciones típicas son la gaussiana, de cauchy, senoide y arco tangente.

En ocasiones, podremos encontrar un **umbral** o límite, en la función de activación, que puede ser entendido como el nivel de excitación de la neurona. Cuando se sobrepasa el valor asignado al umbral, la neurona se activa; y en caso contrario, permanecerá inhibida⁴.

Una vez conocido el funcionamiento de una neurona artificial convendría hacer una descripción de los principales tipos de **arquitectura** que configuran el diseño de un sistema neuronal y los modelos más habituales de **aprendizaje**.

En primer lugar, vamos a entender por arquitectura de una RNA al conjunto de elementos formado por la topología o estructura de conexionado sináptico existente entre sus neuronas, tipo de neurona, número de capas, número de neuronas por capa, etc. Las conexiones entre neuronas o capas pueden dar lugar a redes **unidireccionales**, donde la información fluye en un único sentido; o a redes **recurrentes**, donde la información discurre en cualquier sentido y dirección. El número de capas puede dar lugar a **redes monocapa**, donde sólo existe una única capa o redes **multicapa** [Martín del Brío,1997]. En la figura 2 puede apreciarse una arquitectura de red multicapa con cuatro de ellas, y conexiones unidireccionales.

Un aspecto básico de las RNA reside en su capacidad para **aprender** a partir de ejemplos relacionados con la cuestión planteada. Para ello, la red necesita de un proceso de aprendizaje o entrenamiento, que es iterativo, en el que deberá utilizar un algoritmo que modificará los pesos sinápticos de la red (que reflejan el aprendizaje alcanzado hasta ese momento) hasta que la discrepancia promedio entre sus resultados y los correctos vaya decreciendo y se llegue al rendimiento deseado. Posteriormente, la red deberá ser capaz de asociar a las entradas reales, los resultados que defina el proceso de entrenamiento anterior, es decir, deberá aprender a generalizar.

Para comenzar el proceso de aprendizaje deberá, en primer lugar, elegir la arquitectura de red más adecuada a nuestros propósitos, para, seguidamente, establecer los pesos sinápticos iniciales, que suelen ser valores aleatorios o nulos. A continuación habrá que escoger el algoritmo de aprendizaje más apropiado (existen más de cien); y en último lugar, comenzar con el entrenamiento de la red.

Los tipos de aprendizaje más utilizados y extendidos son el **aprendizaje supervisado** y el **aprendizaje sin supervisión**.

En el aprendizaje **supervisado** se presentan unos patrones de entrada asociados a la salida objetivo; la red deberá ir ajustando sus pesos hasta que la salida sea la deseada. La información de los errores cometidos refuerza los pesos cuando estemos ante clasificaciones correctas y los inhibe en caso contrario. En el **no supervisado** se

⁴ Por ejemplo, si el umbral se sitúa en 0,5, todo valor de la función que supere dicha cifra se convertirá en uno. Por el contrario, el que no la alcance la hará cero.

presenta la entrada pero no la salida deseada. El sistema deberá clasificar las entradas y salidas en base a su similitud con otras entradas. Este tipo de aprendizaje, contiene reglas menos complejas que el anterior aunque su precisión también es menor. Su utilidad se aprecia cuando no sabemos qué salida deberá asociarse a una entrada determinada.

Una vez terminado el proceso de entrenamiento y alcanzado el aprendizaje, el sistema queda preparado para recibir y procesar entradas reales del entorno, con objeto de obtener los resultados que puedan ayudar a resolver el problema considerado.

Por tanto, según la descripción realizada hasta el momento, y en función del diseño que se haya hecho de un sistema neuronal determinado (tipo de neurona, arquitectura de conexiones y algoritmo de aprendizaje utilizados), podremos encontrar diferentes y numerosos modelos que resumimos en los siguientes:

- * Modelo de McCulloch-Pitts
- * Perceptrón de Rosenblatt
- * Adaline (ADaptative LINear Element) de Widrow
- * Matriz adaptativa de Steinbuch
- * Perceptrón multicapa o *backpropagation*
- * Cognitron de Fukushima
- * Redes de Regresión Generalizada de D. Specht: GRNN (*General Regression Neural Network*),
- * Máquina de Boltzmann,
- * Correlación en cascada
- * Teoría de la resonancia adaptativa de Grossberg y Carpenter: ART 1, 2, 3, Fuzzy Art, Artmap, Fuzzy Artmap.
- * Red de Hopfield,
- * Mapas autoorganizativos de Kohonen,
- * Neocognitrón,
- * Contrapropagación de Hecht-Nielsen

Quizás el modelo más popular y utilizado (también en finanzas) sea el **perceptrón multinivel o multicapa** (*Multilayer Perceptron*) de Rosenblatt, entre otros motivos, por considerarse un aproximador universal de funciones:

Consta de múltiples capas interconectadas entre sí, aunque no existen conexiones entre elementos de la misma capa. La información viaja de forma unidireccional desde la entrada, pasando por las capas ocultas y terminando en la salida. Únicamente durante el proceso de aprendizaje la información puede ir hacia atrás con intención de ajustar las ponderaciones asignadas inicialmente a las neuronas (aleatorias con valores pequeños: positivos o negativos). Utiliza el algoritmo de aprendizaje LMS (*Least Mean Square*) de Widrow o el de **retropropagación de errores** (*backpropagation*) de Werbos. Este último algoritmo -el más divulgado-, parte de una muestra de ejemplo y calcula el error de salida. A partir de dicho error actualiza los pesos de salida y, posteriormente, propaga hacia atrás los errores a través de las sinapsis. De este modo, se obtienen las señales de error que corresponden a las capas ocultas y se pueden actualizar sus sinapsis.

Habitualmente, se utiliza al mismo tiempo una muestra de entrenamiento y otra de test para entrenar y validar la red, hasta llegar al punto óptimo, en un proceso denominado validación cruzada. Este proceso es iterativo y cuando termina, los pesos sinápticos quedan fijos y el comportamiento de la red ya no cambia hasta que vuelva a ser reentrenada.

La arquitectura más habitual de este modelo de red: una capa oculta, función de entrada para las salidas de forma lineal, y función de activación sigmoideal.

2. ¿CÓMO DISEÑAR UNA APLICACIÓN?

El diseño de una red neuronal presenta similitudes con los procedimientos empleados en la programación clásica de aplicaciones informáticas, aunque también posee sus especificidades. A continuación y siguiendo las clasificaciones propuestas por Martín del Brío y F. Casacuberta podríamos establecer las siguientes etapas:

1) Plantear o definir el problema a resolver

Tendremos que definir con claridad qué problema queremos resolver, con qué información contamos, y cómo se estructura esta última. Ello nos permitirá reconocer si la técnica más recomendable para la totalidad de nuestro problema es la de las RNA; si puede aplicarse a alguna parte del mismo; o, por el contrario, es conveniente rechazar su aplicación.

2) Elegir el sistema neuronal

Una vez identificado el problema propuesto convendrá elegir el modelo de red⁵, los datos de entrada y número de éstas, las salidas, y el algoritmo de aprendizaje que se adapten mejor a las características observadas en el mismo. En cuanto a las entradas a considerar, tanto la elección de muchas como de pocas variables puede contribuir a una débil generalización. Lo aconsejable es una selección representativa, medida y adecuada.

3) Preprocesamiento

Los datos de entrada y salida de una red neuronal suelen ser heterogéneos en cuanto a su distribución, variabilidad y rangos de valores. Para conseguir que las entradas y salidas sean relevantes en función de su significación real, y ello no dependa de la forma de presentación al sistema neuronal, se realiza un preprocesamiento de los datos. Ello consiste en la transformación de las entradas y salidas de la red, para que puedan ser tratadas de manera adecuada por el sistema. Normalmente, se suelen estandarizar los datos, haciéndose un cálculo de media cero y varianza uno para todos ellos, y acotándolos en un mismo intervalo (por ejemplo (0,+1)), que suele coincidir con el de las funciones de activación empleadas en las capas ocultas y de salida de la red.

⁵ Las arquitecturas más utilizadas son, como ya se indicó anteriormente, el perceptrón multicapa y, también, los mapas autoorganizados de Kohonen.

4) Fase de aprendizaje y test de generalización

En la fase de aprendizaje se realiza el procesamiento de los datos de la fase anterior con ayuda de un algoritmo. Para ello, deben suministrarse al sistema los ejemplos necesarios que permitan describir las situaciones que se presentarán posteriormente en la realidad. En ciertos modelos, como el perceptrón multicapa, se utiliza normalmente una muestra de datos para el aprendizaje de la red, y otra similar para el test de comprobación del nivel de aprendizaje.

Esta etapa terminará cuando los errores de generalización sean mínimos y el modelo se adapte al problema presentado. Para ello, es conveniente probar diferentes topologías y algoritmos de aprendizaje hasta encontrar los más adecuados.

Finalmente, si una vez terminadas las etapas precedentes, no se obtienen las salidas deseadas por el creador de la red, sería conveniente revisarlas, ya que puede que el problema no consiga resolverse por esta técnica, o que las entradas no sean las más adecuadas, o no hayan sido bien preprocesadas, o que las muestras de ejemplo no sean representativas, o que el modelo de red no sea conveniente, etc.

5) Fase de operación

La red se halla en disposición de aplicar el conocimiento adquirido sobre casos nuevos no presentados hasta el momento. Así podremos evaluar si los resultados se mantienen con márgenes de error aceptables. Si ello ocurre el modelo puede ser empleado para solucionar nuestro problema.

3. ¿CUÁNDO ES ÚTIL LA APLICACIÓN DE RNA?

Una vez conocidos de forma genérica los fundamentos que definen a las RNA sería interesante descubrir en qué ámbitos o para qué problemas su utilización podría constituir una técnica más adecuada que otros procedimientos convencionales. Al mismo tiempo intentaremos mostrar los aspectos negativos que desaconsejan su aplicación.

En cuanto a los aspectos favorables:

- ◆ Las RNA no requieren un conocimiento muy profundo para ser formalizadas. Son apropiadas en ámbitos donde el conocimiento es escaso, donde existe información con ruido, imprecisa, incoherente, difusa o borrosa, incierta, contradictoria y errónea, con gran número de entradas y variables correlacionadas, hechos todos característicos de las aplicaciones financieras. Son, por ello, útiles cuando queremos interpretar relaciones entre una gran cantidad de variables y no conocemos con mucha precisión sus reglas de dependencia ni las del problema. Ello nos va a permitir modelizar las funciones de un sistema del que no percibimos su funcionamiento interno y que no exhibe fuertes relaciones fundamentales ni definidas entre sus variables.

- ◆ Las redes neuronales permiten, mediante el uso de muestras históricas, detectar, con más precisión, relaciones no lineales y dependencias temporales. Por este motivo, las verdaderas relaciones y dependencias en los mercados de capitales que, en definitiva, determinan las rentabilidades esperadas de los títulos, serán mejor diseñadas por RNA que a través de, por ejemplo, modelos simples de regresión mínimo cuadrática.
- ◆ Toleran fallos frente a situaciones de ruido o errores en el propio sistema.
- ◆ Generalizan respuestas a un problema, de forma inductiva, a partir de casos reales o ejemplos de un proceso, por lo que la red podrá responder a estímulos no vistos anteriormente. Consecuentemente, las RNA aportan cierta ventaja cuando disponemos de gran cantidad de ejemplos.
- ◆ Son adaptables a situaciones cambiantes, es decir, “aprenden”, ya sea generalizando o reentrenando nuevamente la red. De esta forma, cada vez que cambien las condiciones iniciales del problema no habrá que cambiar de aplicación.
- ◆ Suministran una respuesta rápida a un problema, pudiendo actuar en tiempo real. Ello se debe, entre otros motivos, a que toda la información se procesa en paralelo, de una sola vez.
- ◆ Debido a que el éxito de los métodos de pronóstico depende ampliamente de las propiedades de las series temporales a las que se aplican, los análisis exploratorios de datos anteriores al proceso se hacen muy útiles para obtener mejores pronósticos con modelos como perceptrón multicapa.
- ◆ Por todo lo anterior, las RNA permiten crear sistemas y procesos altamente complejos y no lineales.
- ◆ Finalmente, es destacable la sencillez en su utilización.

Refiriéndonos a sus desventajas:

- * Es difícil explicar la lógica de las RNA en ciertas decisiones. La justificación de los resultados es difícil de mantener, debido a que, por ejemplo, la conexión de los pesos no tiene normalmente interpretaciones obvias. La mayoría de RNA limita la facilidad de explicación, ya que adoptan un enfoque de “caja negra”. Por ello existen dificultades para analizar las características contenidas en cada grupo de observación y la contribución de cada parámetro de entrada sobre el parámetro de salida.
- * El sistema puede hacer que un factor irrelevante para nuestro problema sea importante en el proceso.
- * Es un método iterativo, de prueba y error, por lo que requiere mucha información y tiempo de procesamiento.
- * Pueden no garantizar una solución óptima.
- * No son adecuadas cuando los ejemplos disponibles son escasos.
- * Las RNA todavía no están muy desarrolladas, tanto en sus fundamentos (no tienen un cuerpo formal lógico y muy determinado) como en las aplicaciones realizadas.

4. APLICACIONES DE LAS RNA

Las Redes Neuronales Artificiales se están aplicando actualmente en la solución de problemas procedentes de cualquier ámbito científico. De ellos cabría destacar los siguientes:

1. **Clasificación de datos y reconocimiento de patrones** (fraudes con tarjetas de crédito; reconocimiento de caracteres impresos, manuscritos, o del habla; lucha contra el tráfico de drogas; etc.).
2. **Gestión y proceso de imágenes**
3. **Robótica y control de sistemas**
4. **Aplicaciones en hospitales** (identificación de bacterias, predicción de diversos tipos de cáncer)
5. **Aplicación en Administraciones Públicas** (predicción de la evolución de diferentes indicadores: precios, demandas,...)
6. **Aplicaciones militares** (guiado automático de misiles, combate aéreo,...).
7. **Aplicación en la empresa** (predicción de crisis empresariales, selección de órdenes de compra; análisis de mercados, consumidores, clientes; selección de empleados, etc.).

Ahora bien, de todos los campos de aplicación, el que consideramos más relevante en nuestro estudio es el **financiero**, por ser la disciplina objeto de nuestra consideración. Dentro de este campo se ha trabajado en aplicaciones como:

- Predicción del comportamiento de las acciones en el mercado de valores basada en el análisis técnico y el fundamental.
- Predicción de tipos de interés.
- Análisis de carteras.
- Evaluación y proceso de créditos (*credit scoring*, valoración de hipotecas,...).
- Calificación de obligaciones.
- Análisis de mercados de capitales.
- Modelos de riesgo para inversiones
- Negociación de *commodities* e índices de futuro de renta variable.
- Predicción de la volatilidad de opciones

En definitiva, en un campo de estudio tan amplio como el relacionado anteriormente, nuestro interés, se va centrar específicamente en el **análisis y predicción de las cotizaciones de los títulos de renta variable**, mediante la aplicación de redes neuronales, y como alternativa a los procedimientos clásicos, estadísticos o de cualquier otro tipo que puedan utilizarse en su medición.

5. SISTEMAS NEURONALES EMPLEADOS EN LOS MERCADOS FINANCIEROS DE RENTA VARIABLE

A continuación ofrecemos las características generales de una serie de trabajos que han intentado presentar soluciones a diversos temas relacionados con la clasificación y predicción en el campo financiero. En ellos, la utilidad que tiene la aplicación de la técnica de las redes neuronales, parte de la idea de que es posible determinar regularidades no detectadas en los movimientos de precio de los activos

financieros. Estas regularidades se constituyen en ineficiencias⁶ temporales del mercado financiero, que pueden explotarse sistemáticamente a corto plazo, y que en el largo plazo tienden a desaparecer.

De este modo, el movimiento de los precios no es aleatorio, se pueden observar formas no lineales de evolución; incluso hay quien advierte que las series temporales de precios son consecuencia de un proceso dinámico altamente no lineal, cuyas ecuaciones desconocemos. Estas ideas se alejan de la teoría de los mercados eficientes o paseo aleatorio, que afirma que el movimiento de los precios de los activos no es predecible, pues sigue una marcha incierta.

En definitiva, se trata de poner al alcance del inversor una tecnología que permita procesar más rápidamente el conocimiento humano sobre los mercados financieros, y que pueda aprovechar los beneficios que ella aporta, a partir de la determinación de procesos no lineales y caóticos (muy sensibles a las condiciones iniciales) y la explotación de ineficiencias a corto plazo.

5.1. Principales casos tratados

Hemos encontrado estudios que intentan descubrir ineficiencias en los mercados financieros en White (1988), Tsiburis y Zeidenberg y Bosarge, Jr. (1991). Autores como Kryzanowski, Galler y Wright (1993), parten de la evidencia de que las rentabilidades a corto plazo pueden ser pronosticadas a partir de los fundamentos de la compañía.

Otros estudios profundizan en el intento de obtener modelos que puedan reportar **mayor rentabilidad** que la lograda en los mercados financieros. De esta forma, Baestaens y Van den Bergh analizan la rentabilidad de la Bolsa de Amsterdam con el objetivo de descubrir las fuerzas que marcan los cambios direccionales de las rentabilidades. En una línea similar, aunque con otro tipo de información, trabajaron Steiner y Wittkemper, así como Yoon y Swales (1991). Para finalizar, Kimoto, Asakawa, Yoda y Takeoka (1990) pretendían determinar el momento óptimo para la compraventa del índice japonés TOPIX.

Por otro lado, encontramos estudios basados en el tratamiento de series temporales para el pronóstico financiero, como en Sharda y Patil (1992), y Jhee y Shaw; y trabajos sobre los mercados financieros primarios y derivados en Jain y Nag, y Trippi y DeSieno (1992), respectivamente.

Así, pues, basándonos en estos trabajos, vamos a presentar las peculiaridades que más habitualmente se repiten en los modelos neuronales financieros observados. De este modo, veremos las principales entradas, arquitecturas, algoritmos de entrenamiento y salidas de los sistemas neuronales, lo que nos ayudará a fijar las estructuras de más frecuente utilización en la predicción financiera.

⁶ La ineficiencia básica buscada por el inversor es encontrar diferencias de valor o precio entre títulos que teóricamente deban tener el mismo comportamiento.

5.2. Entradas

En referencia a las **entradas** que se emplean en las redes neuronales observadas hemos detectado que esencialmente la información es de tipo fundamental o técnico.

En cuanto a los datos **técnicos** encontramos *rentabilidades diarias*, en White, y Tsiburis y Zeidenberg; *precios y estadísticos derivados de precios pasados*, en Trippi y DeSieno; y *volúmenes, precios diarios y rentabilidades de índices*, en Steiner y Wittkemper.

La información de carácter **fundamental** que se presenta a los sistemas neuronales se refiere a *datos financieros de empresas sobre rentabilidad, deuda, liquidez y actividad; información financiera de su sector*, así como datos de *factores macroeconómicos*, en Kryzanowski, Galler y Wright; *información económica primaria* con influencia en la valoración del título (*tamaño de la emisión, reputación del emisor,...*), en Jain y Nag; *indicadores macroeconómicos* como *tipo de interés, inflación y tipo de cambio*, en Baestaens y Van den Bergh; e Hiemstra y Haefke; e *información cualitativa* dirigida a los accionistas (*confidencias, factores económicos fuera del control de la empresa, crecimiento, planes estratégicos, nuevos productos,...*), en Yoon y Swales

5.3. Salidas

Las **salidas** del sistema tienen que ver con los objetivos de predicción y clasificación perseguidos. Así, encontramos entre otros conocer el *precio de la acción o del índice*, en White; Kimoto, Asakawa, Yoda y Takeoka; y Jain y Nag; las *rentabilidades de títulos o índices*, en Tsiburis y Zeidenberg; Baestaens y Van den Bergh; e Hiemstra y Haefke; obtener una *recomendación, larga o corta, de inversión*, en mercados de futuros en Trippi y DeSieno; y estimar la *tendencia o comportamiento del precio de los títulos*, en Kryzanowski, Galler y Wright; y Yoon y Swales.

5.4. Arquitecturas y algoritmos

Se observa que en casi la totalidad de autores la topología de red más utilizada corresponde al perceptrón multicapa de transmisión unidireccional con una capa de entrada, una oculta y una de salida, y con el algoritmo de entrenamiento *backpropagation*. Como funciones de activación o transferencia se usaron, sobre todo, la sigmoideal, tangente y lineal. Además, en un caso se vio una topología de dos capas ocultas y se utilizó el método de aprendizaje supervisado de la máquina de Boltzmann. Por su parte, los modelos comparados fueron modelos estadísticos de regresión y de análisis discriminante multivariante.

6. APLICACIÓN PRÁCTICA DE LAS REDES NEURONALES AL MERCADO FINANCIERO ESPAÑOL

A continuación, basándonos, por una parte, en los estudios empíricos desarrollados por diversos autores y, por otro lado, en nuestra propia experiencia sobre mercados de valores, vamos a intentar realizar una estimación de rentabilidades del

mercado de capitales español. Nuestro propósito tiene un carácter meramente ejemplificativo; trata de plantear de forma práctica como se realiza el diseño, en todas sus etapas, de una predicción financiera mediante redes neuronales y los resultados obtenidos. En ningún caso, intentamos proponer el mejor diseño y funcionamiento de red. Nuestras pretensiones consisten simplemente en comprobar la oportunidad de utilizar las redes neuronales en la predicción financiera observando sus resultados, que, en todo caso, pueden y deben ser mejorables.

Así pues, para describir el diseño de red utilizada y aplicada, seguiremos los pasos desarrollados en el punto 2 del presente trabajo, comenzando por la definición del problema.

1) Planteamiento del problema a resolver

En un primer momento nuestra intención se centraba en intentar hacer una predicción, de uno a cinco días, del valor de cierre alcanzado por el índice del mercado español IBEX-35. Como quiera que los resultados obtenidos no estaban dentro de los márgenes que considerábamos aceptables, decidimos proponer como objetivo la predicción de la rentabilidad (R_t) de dicho índice de uno a cinco días expresada como:

$$R_t = \frac{I_t - I_{t-1}}{I_{t-1}}$$

Donde, I_t es la cotización de cierre del IBEX-35 en el día t observado, I_{t-1} , la cotización de cierre del día anterior, e $I_t - I_{t-1}$, el incremento o retroceso del IBEX en el período $t-1, t$.

La información utilizada para la predicción de nuestro problema abarca un horizonte temporal que se inicia en 1995 y finaliza en 1998. Para ello, y en base a nuestra experiencia, propusimos las variables, tanto de orden técnico, fundamental o estadístico que podrían ofrecernos una mayor correlación con la variable objetivo estimada. De este modo las variables consideradas fueron:

- ✓ La cotización de cierre del Bono Nocional a 10 años del mercado de Deuda Pública español.
- ✓ La cotización de cierre al contado del tipo de cambio euro/dólar.
- ✓ La cotización de cierre del día anterior al cierre del IBEX del índice norteamericano Dow-Jones.
- ✓ El Índice de Fuerza Relativa (RSI) del IBEX-35.
- ✓ El indicador estocástico del IBEX-35.
- ✓ Y, finalmente el indicador Acumulación/Distribución del IBEX-35.

2) Elección del sistema neuronal

En primer lugar, el tipo de **software** empleado en el estudio ha sido *ThinsPro Neural Network for Windows Special Edition* Versión 1.00. Las limitaciones de dicho software nos hicieron ser cautelosos a la hora de afrontar un diseño demasiado complejo, por lo que en cierto modo enfocamos un estudio que no contuviera excesivas entradas, nodos ocultos y datos de observación.

En este sentido, las **entradas** originales (sin transformación) que se presentaron al sistema fueron un total de cinco:

- Bono nocional a 10 años
- Tipo de cambio euro/dólar
- Índice Dow-Jones
- RSI del Ibex-35
- Estocástico del Ibex-35

Además, también se empleó, en algún caso, la media móvil a cinco días de cada una de las variables anteriores. De este modo, las entradas al sistema fueron efectuadas con sus datos originales, sin ninguna modificación, o bien combinadas con las medias móviles calculadas a cinco días. Así, se crearon múltiples diseños, en donde se combinaban variables de entrada originales con medias móviles, o en los que simplemente se consideraban las primeras o las últimas de forma separada. La presentación de los datos se hizo en el orden temporal de cada período considerado, si bien también probamos, en otros casos, una introducción aleatoria de los mismos.

Como **salida** del sistema se seleccionó, en principio -como ya se comentó anteriormente-, la cotización de cierre del índice español Ibex-35; no obstante, los malos resultados obtenidos nos hicieron optar por un cambio hacia la estimación de la rentabilidad diaria de dicho índice.

Una vez determinadas y elegidas las entradas y salidas al sistema, consideramos adecuado un **período de entrenamiento** de dos años, desde 1-1-1995 hasta 31-12-1996; así como un **período de test** de la misma duración, desde 1-1-1997 hasta 31-12-1998. No obstante en algunos diseños optamos por reducir los datos del test al 15% de los utilizados en el entrenamiento.

En los períodos de entrenamiento y test eliminamos aquellos días en los cuales no existían observaciones para alguna de las variables. De esta forma, dejamos, en cualquier día, solo los datos que tenían valor para todas las variables del estudio. En el caso del Dow Jones se asoció el valor de cierre del día anterior con los valores del resto de variables del día posterior.

En cuanto a la arquitectura utilizada correspondió a una perceptrón multicapa (*Multilayer Normal Feed Forward*) con regla de entrenamiento de retropropagación de errores (*backpropagation*) y con una capa oculta (*hidden layer*) de ocho y diez neuronas, en el caso de usar cinco entradas; o quince, en caso de utilizar un máximo de diez entradas. Los pesos se inicializaron en torno al 0,3 – 0,6 y los ratios de aprendizaje se situaron en 1,5 en la capa oculta y 0,01 en la de salida.

3) Preprocesamiento

Como **función de preprocesamiento** (*preprocessor functions*) se aplicó la desviación estándar media (*mean/std deviation function*). La **función de entrada** (*input function*) utilizada, tanto para los nodos ocultos como para el de salida, fue *Dot Product* que representa la suma ponderada de los pesos de cada neurona por las entradas a la misma; siendo la **función de transferencia** (*transfer function*) para los nodos ocultos

arcotangente bipolar (entre -1 y $+1$), excepto para la salida que fue la de seno bipolar ($-1,+1$). La tolerancia de error para las salidas del sistema (rentabilidades diarias) con respecto a las salidas deseadas se situó en $0,2\%$, siendo el error tipo empleado por la red el cuadrático medio (*Mean Square Error*).

4) Fase de aprendizaje y test de generalización

Con referencia al número de observaciones, hubo casos en los que las muestras utilizadas en el entrenamiento fueron de hasta 750 (máximo admitido por la red) y en otros de unas 478, correspondiendo a dos años de datos. Por su parte, en el test hubo diseños con dos años de observaciones y otros con un 15% de los datos usados en la fase de entrenamiento.

Con este diseño e información, la red fue detenida utilizando la validación cruzada, justo cuando las prestaciones de la red aplicadas al test empezaban a empeorar.

A partir de aquí los **resultados** obtenidos por los diseños practicados no son satisfactorios. El error cometido en el entrenamiento es muy amplio. Después de realizar múltiples variaciones en el diseño: elegir menos entradas; cambiar las funciones de transferencia y entrada; o cambiar de algoritmo; los errores cometidos seguían siendo altos.

Paradójicamente el diseño que mejores resultados ofreció fue aquél que empleaba las cinco variables originales con el valor de cierre del mismo día que estimaba, de modo que el error de entrenamiento se situó en un $0,0019$, el de test en $0,0098$, las respuestas de la red que quedaban dentro del margen de error fueron en el entrenamiento del $78,41\%$ y en el test del $38,22\%$. Para el resto de diseños y combinaciones todos los resultados fueron peores que los reseñados anteriormente, por lo cual después de horas y horas de prueba decidimos terminar nuestra labor, sin haber encontrado en nuestro ámbito de estudio buenos resultados aplicando la teoría de las redes neuronales.

A la vista de los resultados obtenidos en todo el estudio, si bien, como comentamos más arriba, se trata simplemente de una experiencia particular y no generalizable, hemos creído oportuno seguir profundizando en otras técnicas, con más contenido lógico y racional. No dudamos que las redes neuronales puedan ser aplicadas con éxito en otros campos científicos, incluso afines al nuestro; sin embargo, tenemos nuestras reservas en su aplicabilidad directa a nuestro ámbito.

BIBLIOGRAFÍA:

- Apostolos y Refenes, P. (1995): *Neural Networks in the Capital Markets*, London Business School, England, A Wiley Finance Edition.
- Baestaens, D. y Van den Bergh, W.: “Tracking the Amsterdam Stock Index Using Neural Networks”, publicado en *Neural Networks in the Capital Markets*, London Business School, England, A Wiley Finance Edition, 1995.
- Bosarge, W. (1991): “Adaptive Processes to Exploit the Nonlinear Structure of Financial Markets”, Santa Fe Institute of Complexity Conference; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1992.
- Casacuberta, F. (1998): “Redes Neuronales Artificiales”, *Novática*, nº131, enero-febrero, pp. 15-17.
- Cox, E. (1993): “Adaptative Fuzzy Systems. (Advance Technology & Controls)”. *IEEE Spectrum*, pp.27-31
- De la Fuente, D. y Pino, R. (1996): “Aplicación de sistemas expertos y redes neuronales en organización...”, *Esic Market*, octubre-diciembre, pp-173-179.
- Forrester, J.W. (1998): “Diseñando el futuro”, en *Lección Magistral para la investidura como Doctor Honoris Causa por la Universidad de Sevilla*, pp.19-30.
- Gately, E. (1996): *Neural Networks for Financial Forecasting*, John Wiley & Sons, Inc., New York.
- Gento, A.M. y Del Olmo, R. (1994): “Redes Neuronales Fuzzy en la gestión de empresas”, *Anales de Estudios Económicos y Empresariales*, pp. 37-71.
- Hetch-Nielsen, R (1988): “Neurocomputing: picking the human brain”, *IEEE Spectrum*, March.
- Hiemstra, Y. y Haefke, C.: “Two Multilayer Perceptron Training Strategies for Low-Frequency S&P 500 Prediction”, publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.

- Jain, B. y Nag, B.: "Artificial Neural Network Models for Pricing Initial Public Offerings", publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.
- Kimoto, T., Asakawa, K., Yoda M., y Takeoka M. (1990): "Stock Market Prediction System with Modular Neural Networks", *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. I1-I6; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.
- Kryzanowski, L., Galler, M., y Wright, D.(1993): "Using Artificial Neural Networks to Pick Stocks", *Financial Analyst Journal*, July/August; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996
- Márquez, L., Hill, T., Worthley, R., y Remus, W. (1991): "Neural Network Models as an Alternative to Regression", *Proceedings of the IEEE 24th Annual Hawaii International Conference on Systems Sciences*, vol. VI, pp.129-35; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.
- Martín, B. y Sanz, A. (1997): *Redes Neuronales y Sistemas Borrosos*, Editorial Ra-Ma, Madrid.
- Moreno García, A. (1998): *Los Sistemas Expertos y su aplicación a la gestión empresarial*. Tesis doctoral. Departamento de Economía Financiera y Dirección de Operaciones. Universidad de Sevilla.
- Sigüenza, J.A., López, V., y Dorronsoro, J.R. (1993): "Redes Neuronales, de la investigación básica a las aplicaciones industriales, *Chip*, Abril, pág.66.
- Sellés, M. (1995): "Las redes neuronales: un sistema de computación de múltiples aplicaciones para las empresas", *Novática*, nº114, marzo-abril, pp.73-79.
- Serrano, C. y Gallizo, J.L. (1996): "Las redes neuronales artificiales en el tratamiento de la información financiera", <http://ciberconta.unizar.es/Biblioteca/0004/SerGall96.html>
- Sharda, R. y Patil, R. (1992): "A Connectionist Approach to Time Series Prediction: an Empirical Test", *Journal of Intelligent Manufacturing*; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.
- Steiner, M. y Wittkemper, H.: "Neural Networks as an Alternative Stock Market Model", publicado en *Neural Networks in the Capital Markets*, London Business School, England, A Wiley Finance Edition, 1995.
- Trippi, R. y DeSieno, D. (1992): "Trading Equity Index Futures with a Neural Network", *Journal of Portfolio Management* 18, nº15, pp.27-33; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.

- Trippi, R. y Turban, E. (1992): *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real-World Performance*, Irwin Professional Publishing, New York.
- (1996): *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real-World Performance*, Irwin Professional Publishing, New York.
- Tsiburis, G. y Zeidenberg, M.: “Testing the Efficient Markets Hypothesis with Gradient Descent Algorithms”, publicado en *Neural Networks in the Capital Markets*, London Business School, England, A Wiley Finance Edition, 1995
- White, H. (1988): “Economic Prediction using Neural Networks: The case of IBM Daily Stock Returns”, *Proceedings of the IEEE International Conference on Neural Networks*, pp. II-451-II-458; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.
- Yoon, Y. y Swales, G. (1991): “Predicting Stock Price Performance: a Neural Network Approach”, *Proceedings of the IEEE 24th Annual International Conference on Systems Sciences*, pp.156-62; publicado en *Neural Networks in Finance and Investing*, Irwin Professional Publishing, New York, 1996.