

APLICACIÓN DE REDES NEURONALES ARTIFICIALES A LA PREVISIÓN DE SERIES TEMPORALES NO ESTACIONARIAS O NO INVERTIBLES

R. PINO*

D. DE LA FUENTE

J. PARREÑO

P. PRIORE

Universidad de Oviedo

En los últimos tiempos se ha comprobado un aumento del interés en la aplicación de las Redes Neuronales Artificiales a la previsión de series temporales, intentando explotar las indudables ventajas de estas herramientas. En este artículo se calculan previsiones de series no estacionarias o no invertibles, que presentan dificultades cuando se intentan pronosticar utilizando la metodología ARIMA de Box-Jenkins. Las ventajas de la aplicación de redes neuronales se aprecia con más claridad, cuando se trata de pronosticar sistemas multivariantes no estacionarios.

The application of artificial neural networks to forecasting non-stationary or non-invertible time series

Palabras clave:

Clasificación AMS (MSC 2000): 37M10 Time Series Analysis
62M10 Time Series
62M45 Neural Networks
82C32 Neural Networks

*Dpto. Admon. de Empresas y Contabilidad. Escuela Politécnica de Ingenieros de Gijón. Campus de Viesques, s/n. 33204 GIJÓN (Spain). E-Mail: pino@etsiig.uniovi.es.

–Recibido en mayo de 2001.

–Aceptado en octubre de 2002.

1. INTRODUCCIÓN

Es un hecho conocido que para que el modelo de una serie temporal ARIMA pueda ser utilizado correctamente, se deben cumplir las condiciones de estacionariedad e invertibilidad (Box and Jenkins, 1970). En los modelos autorregresivos (AR), estas condiciones se cumplen si todas las raíces del polinomio característico están situadas fuera del círculo unidad; la misma condición asegura la invertibilidad de los modelos media móvil (MA). Para los modelos autorregresivos y media móvil (ARMA), existen varias condiciones que garantizan la estacionariedad e invertibilidad del modelo para determinadas localizaciones de las raíces de los polinomios característicos de los factores AR y MA (Huang, 1990; Huang and Anh, 1993).

Si estas condiciones no se cumplen, la metodología de Box-Jenkins llevaría al cálculo de previsiones fuera de unos límites razonables, por lo que, en el caso de series temporales no estacionarias o no invertibles, es necesario realizar transformaciones previas que garanticen estas dos condiciones antes de pasar a la estimación del modelo y al posterior cálculo de las previsiones.

Una de las características fundamentales de las Redes Neuronales Artificiales (RNAs) es su capacidad de «aprender» a partir de los ejemplos que se le proporcionan, sin hacer suposiciones a priori sobre los modelos y relaciones que subyacen en la serie temporal (Zhang *et al.*, 1998). Esta propiedad hace que sea posible calcular previsiones de cualquier serie temporal sin tener la necesidad de asegurar previamente ninguna de las condiciones comentadas anteriormente (estacionariedad e invertibilidad). Por tanto, el tratamiento de los datos, no dependerá de si es estacionaria o no, sino que será el mismo que se lleva a cabo con el resto de las series temporales (normalización de los datos, eliminación de la tendencia, etc.)

Hemos querido demostrar estos hechos, utilizando RNAs para calcular previsiones de varias series temporales no estacionarias o no invertibles. En el apartado 2 se hace una breve introducción a las redes neuronales artificiales, destacando las dos arquitecturas que se utilizan en este estudio; posteriormente se pronostican dos modelos autorregresivos no estacionarios de órdenes 1 y 2 (en los apartados 3 y 4 respectivamente); en el apartado 5 se calculan previsiones de un modelo media móvil no invertible de orden 2. Hasta aquí, todos los sistemas pronosticados son univariantes, pero donde verdaderamente se comprueba la utilidad de las RNAs, es al calcular previsiones de sistemas multivariantes, como se puede comprobar en el apartado 6, en el que se han aplicado las RNAs en el cálculo de previsiones de un sistema multivariante no estacionario. Terminamos el estudio resumiendo las conclusiones alcanzadas.

2. REDES NEURONALES ARTIFICIALES

En este apartado describiremos brevemente la estructura de una red neuronal (especialmente el Perceptrón Multicapa (MLP) y las redes de Elman, que son las que utilizaremos en este estudio), pero antes intentaremos establecer lo que se entiende por Computación Neuronal. Con este término definimos el estudio de estructuras de cálculo paralelo basadas en procesadores elementales adaptables (neuronas) ampliamente interconectados que, a partir del aprendizaje mediante ejemplos, almacenan conocimiento experimental y lo hacen disponible para su uso.

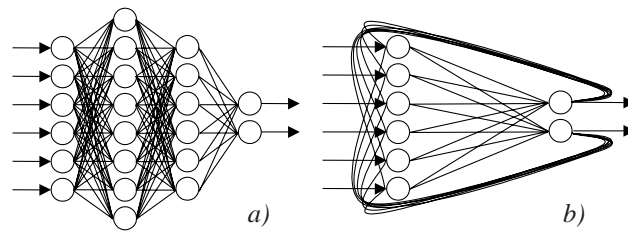


Figura 2.1. Ejemplos de Redes Neuronales «Feedforward» (a) y «Feedback» (b).

El Perceptrón Multicapa (Multilayer Perceptron, MLP), también conocido como Red Backpropagation (Backpropagation Net, BPN) es uno de los modelos de Red Neuronal Artificial más conocidos y utilizado en la práctica como clasificadores de patrones y aproximadores de funciones (Lippman, 1987; Freeman and Skapura, 1991). Pertenecen a la clase de las denominadas redes no realimentadas o «feedforward», su topología es la de un sistema neuronal estructurado en capas (Fig. 2.1.a), donde la información siempre fluye desde la capa de entrada, cuyo único papel es el de enviar los datos de entrada al resto de la red, hacia la de salida, atravesando la capa o capas ocultas. Esencialmente son las capas internas las encargadas de realizar el procesamiento de la información, extrayendo rasgos característicos de los datos de entrada. Aunque existen multitud de variantes, normalmente cada neurona de una capa se conecta a todas las neuronas de la capa siguiente; sin embargo, no existe conexión, ni por tanto interacción, entre las neuronas de una misma capa.

El algoritmo de entrenamiento más habitualmente utilizado en el MLP es el denominado «Retro-propagación» (Backpropagation, BP; Rumelhart and McClelland, 1986; Rumelhart *et al.*, 1986). Se trata de un aprendizaje de tipo *supervisado* en el que se muestra a la red neuronal tanto los patrones de entrada como su salida deseada, la red aprenderá a asociarlos por medio de la regla de aprendizaje, modificando los pesos sinápticos. Mediante este tipo de aprendizaje una red neuronal puede aproximar una compleja función a partir de muestras de ella o clasificar patrones a partir de ejemplos correctamente clasificados.

El BP, pese a ser de lenta convergencia y no garantizar el alcance de un mínimo global de la función de error, es un algoritmo muy utilizado por su relativa simplicidad y eficiencia. Se han propuesto en la literatura muchas variaciones para resolver estos problemas (Freeman and Skapura, 1991). Uno de los más utilizados para acelerar la convergencia y eliminar posibles oscilaciones en torno a mínimos locales, es la adición, a la hora de variar los pesos, de un determinado término denominado *momento*, que tiene en cuenta la anterior variación de los pesos.

El segundo tipo de redes que se utilizan en este estudio son las redes parcialmente recurrentes (Recurrent Neural Networks, RNN). Estas redes tienen una arquitectura basada en el MLP, pero con algunas modificaciones que las hacen especialmente interesantes para aplicarlas a predicción. En ellas existe una capa con neuronas especiales (denominada capa contextual) en la que se guarda la activación del instante anterior de las neuronas de otra capa (la de salida en las Redes de Jordan, o la oculta en las de Elman). En la figura 2.2. se pueden observar las dos estructuras más típicas de redes parcialmente recurrentes (Elman, 1990; Jordan, 1986).

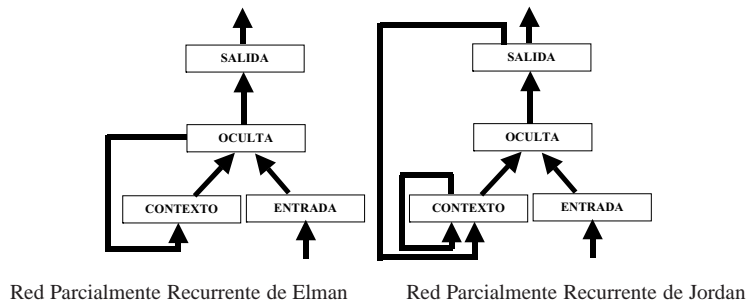


Figura 2.2. Arquitecturas típicas de Redes Parcialmente Recurrentes.

Las recurrencias aparecen, si consideramos que la copia de la activación de la capa de neuronas normales a la capa contextual se realiza a través de unas conexiones con pesos fijos con valor 1. La particularidad de las redes parcialmente recurrentes de poder captar la evolución temporal de una variable en instantes consecutivos las hacen especialmente interesantes y apropiadas para predicción (Debar *et al.*, 1992; Dorffner *et al.*, 1994; Gordon *et al.*, 1991; Lee and Park, 1992; Kamijo and Tanigawa, 1993).

3. MODELO AR(1) NO ESTACIONARIO

Si tenemos un modelo autorregresivo de orden uno:

$$(1 - \phi_1 B) \cdot x_t = \varepsilon_t$$

Para que el proceso sea estacionario, se debe cumplir que el parámetro ϕ_1 satisfaga la condición $|\phi_1| < 1$ (Box and Jenkins, 1970), lo que equivale a decir que la raíz de la ecuación $1 - \phi_1 B = 0$ debe estar situada fuera del círculo unidad (figura 3.1).

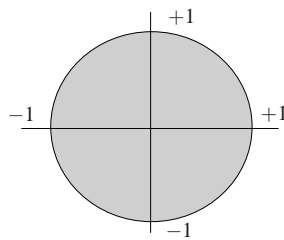


Figura 3.1. Círculo unidad.

Para comprobar la capacidad de las RNAs para pronosticar series temporales no estacionarias, hemos simulado la serie temporal llamada AR1 que tiene el siguiente modelo:

$$(1 - 1,01 \cdot B) \cdot x_t = \varepsilon_t$$

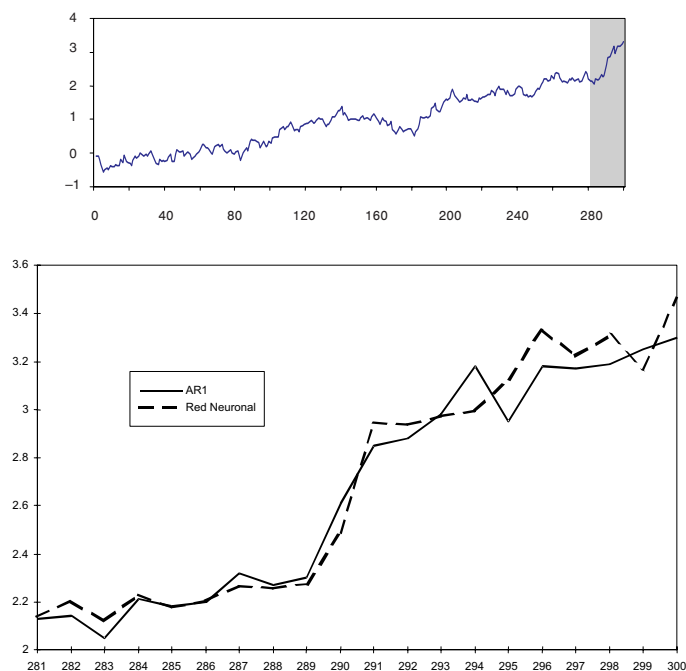


Figura 3.2. Serie AR1 (modelo AR(1) no estacionario).

Se puede comprobar que no se cumple la condición de estacionariedad, pues $\phi_1 = 1,01 > 1$. En la figura 3.2, se ha representado la serie AR1 simulada (la zona sombreada representa la zona que se pronostica), y a su derecha, se muestran las previsiones calculadas empleando un perceptrón multicapa 14-7-1 (14 neuronas en la capa de entrada, 7 en la oculta y 1 en la salida).

4. MODELO AR(2) NO ESTACIONARIO

Supongamos ahora el modelo autorregresivo de orden 2:

$$(1 - \phi_1 B - \phi_2 B^2) \cdot x_t = \varepsilon_t$$

Como ya se ha dicho en el caso anterior, para que el proceso sea estacionario, se debe cumplir que las raíces del polinomio característico estén fuera del círculo unidad, lo cual implica que los parámetros del modelo ϕ_1 y ϕ_2 cumplan las siguientes condiciones (Box and Jenkins, 1970):

$$\begin{cases} \phi_2 + \phi_1 < 1 \\ \phi_2 - \phi_1 < 1 \\ -1 < \phi_2 < +1 \end{cases}$$

Estas tres condiciones se pueden representar gráficamente en unos ejes ϕ_1 y ϕ_2 , de forma que si el punto designado por (ϕ_1, ϕ_2) está situado dentro del triángulo, se puede demostrar que el proceso es estacionario, y si está situado sobre los bordes o fuera del triángulo, el proceso será no estacionario (figura 4.1).

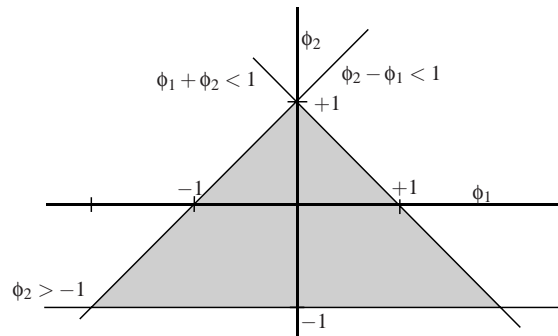


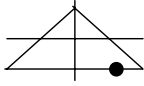
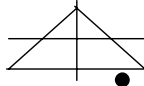
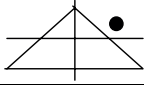
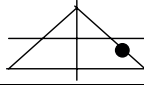
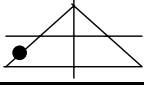
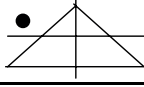
Figura 4.1. La zona sombreada implica estacionariedad.

Hemos simulado seis combinaciones de los parámetros ϕ_1 y ϕ_2 que hacen no estacionario un proceso AR(2), obteniendo las series temporales llamadas AR21, ..., AR26. En

la tabla 4.1, se muestran los valores de los parámetros de las seis series y su situación con respecto al triángulo. Por ejemplo, el modelo de Box-Jenkins de la serie AR21 será:

$$x_t - 1,2 \cdot x_{t-1} + x_{t-2} = \varepsilon_t$$

Tabla 4.1. Series No Estacionarias AR(2).

Serie	Situación	Parámetros	Serie	Situación	Parámetros
AR21		$\phi_1 = 1,2$ $\phi_2 = -1,0$	AR22		$\phi_1 = 1,2$ $\phi_2 = -1,1$
AR23		$\phi_1 = 1,2$ $\phi_2 = 0,5$	AR24		$\phi_1 = 1,2$ $\phi_2 = -0,2$
AR25		$\phi_1 = -1,7$ $\phi_2 = -0,7$	AR26		$\phi_1 = -1,7$ $\phi_2 = 0,5$

En las figuras 4.2 hasta 4.7, se presentan las series y las previsiones que se han calculado para cada una de ellas.

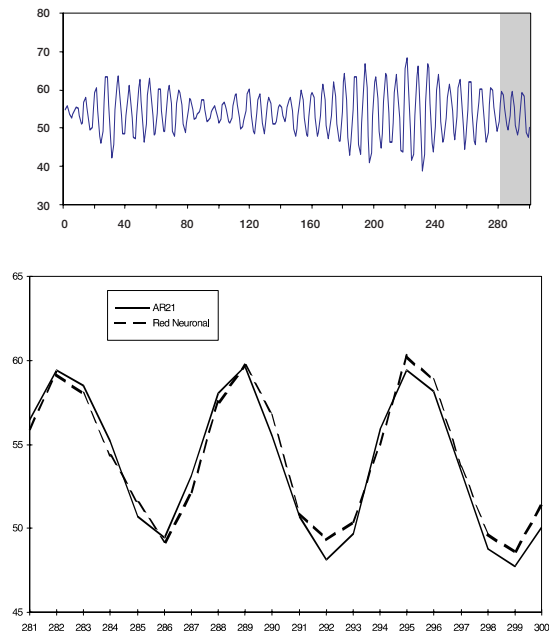


Figura 4.2. Serie AR21 (modelo AR(2) no estacionario).

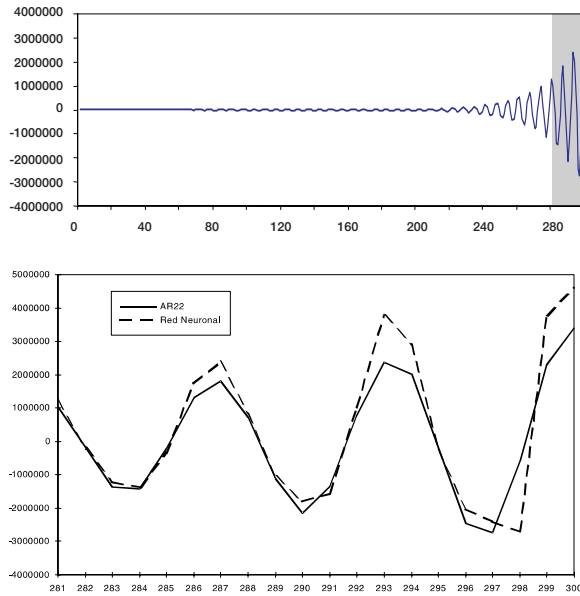


Figura 4.3. Serie AR22 (modelo AR(2) no estacionario).

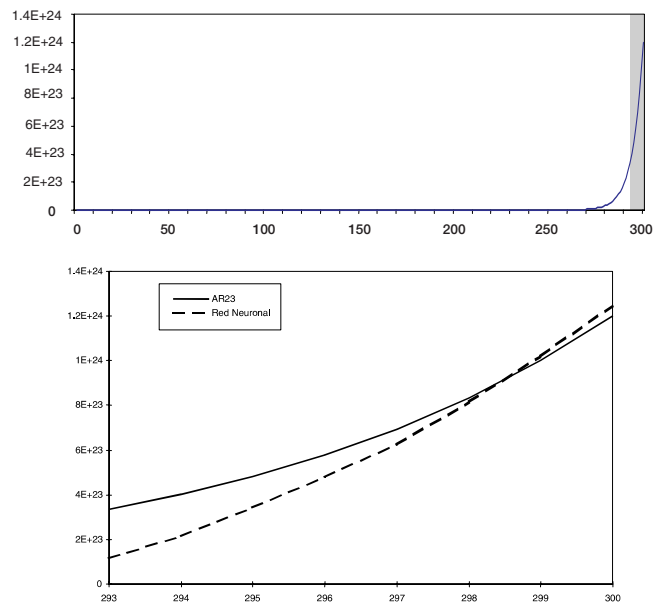


Figura 4.4. Serie AR23 (modelo AR(2) no estacionario).

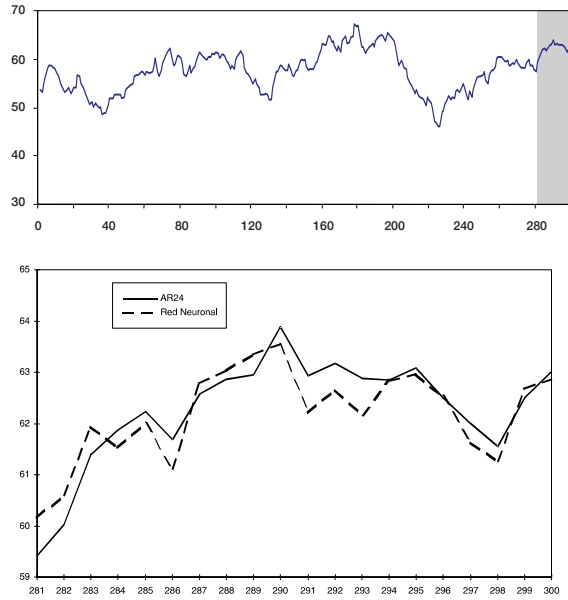


Figura 4.5. Serie AR24 (modelo AR(2) no estacionario).

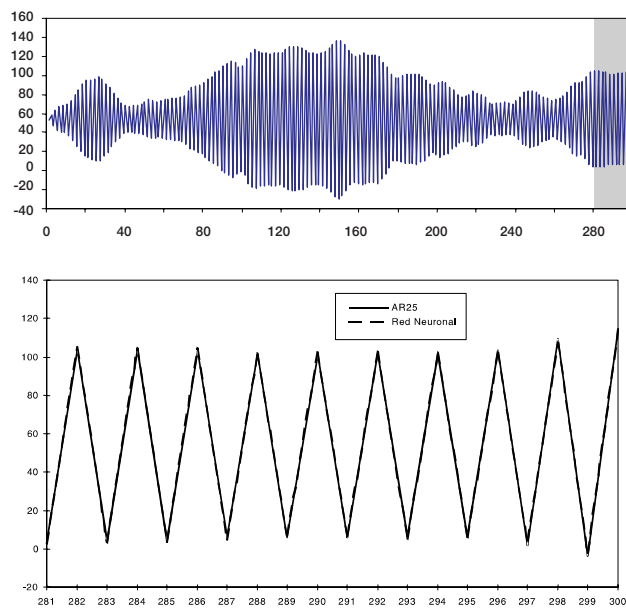


Figura 4.6. Serie AR25 (modelo AR(2) no estacionario).

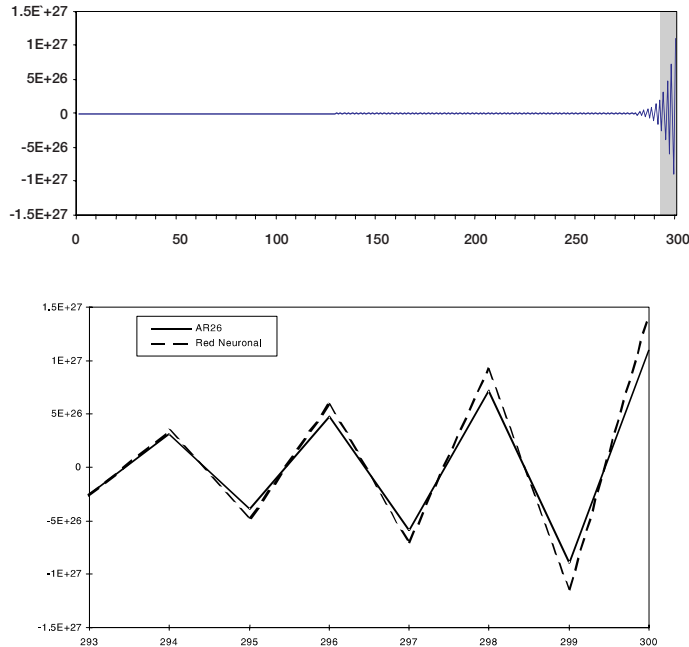


Figura 4.7. Serie AR26 (modelo AR(2) no estacionario).

5. MODELO MA(2) NO INVERTIBLE

Si tenemos ahora un proceso que presenta un modelo Media Móvil de orden 2:

$$x_t = (1 - \theta_1 B - \theta_2 B^2) \cdot \varepsilon_t$$

Las condiciones de estacionariedad de los modelos autorregresivos, aquí se convierten en condiciones de invertibilidad, por lo tanto los parámetros θ_1 y θ_2 deben cumplir (Box and Jenkins, 1970):

$$\begin{cases} \theta_2 + \theta_1 < 1 \\ \theta_2 - \theta_1 < 1 \\ -1 < \theta_2 < +1 \end{cases}$$

Al igual que ocurría en los procesos autorregresivos, se pueden representar gráficamente estas condiciones, de tal manera que si el punto designado por (θ_1, θ_2) está situado dentro del triángulo, el proceso es invertible mientras que si está fuera o sobre el borde, no será invertible (figura 5.1).

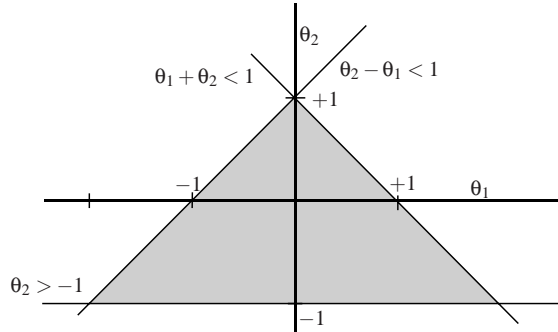


Figura 5.1. La zona sombreada implica invertibilidad.

Hemos simulado las seis combinaciones de los parámetros θ_1 y θ_2 , que hacen no invertible el proceso, obteniendo las series MA20, MA21, ..., MA25. En la tabla 5.1, se muestran los valores de los parámetros, que hemos utilizado para la obtención de cada una de las series, y la situación del punto (θ_1, θ_2) con respecto al triángulo.

Tabla 5.1. Series No Invertibles MA(2).

Serie	Situación	Parámetros	Serie	Situación	Parámetros
MA20		$\theta_1 = -0,5$ $\theta_2 = 1,0$	MA21		$\theta_1 = 0,5$ $\theta_2 = -1,0$
MA22		$\theta_1 = 0,5$ $\theta_2 = -1,5$	MA23		$\theta_1 = 1,6$ $\theta_2 = -0,6$
MA24		$\theta_1 = 2,0$ $\theta_2 = -0,6$	MA25		$\theta_1 = -1,6$ $\theta_2 = -0,6$

En las figuras 5.2 hasta 5.7, se han representado las seis series temporales y las previsiones que se han calculado en cada uno de los casos.

En todas las series anteriores, se han calculado previsiones utilizando el programa de simulación de redes neuronales SNNS 4.2 (Stuttgart Neural Network Simulator). Las arquitecturas que se han probado son el Perceptrón Multicapa (MLP) y las Redes Parcialmente Recurrentes de Elman (ELM). En cada caso se ha seguido un proceso de prueba y error para determinar el número adecuado de neuronas tanto en la capa de entrada como en la capa oculta.

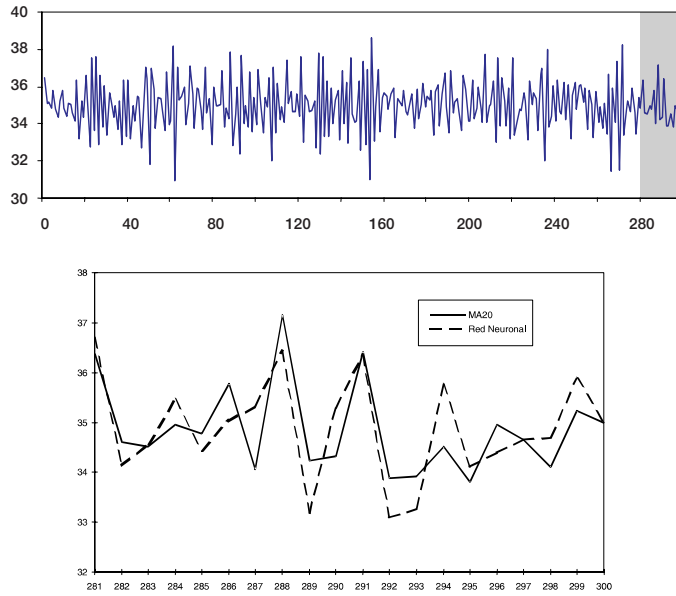


Figura 5.2. Serie MA20 (modelo MA(2) no invertible).

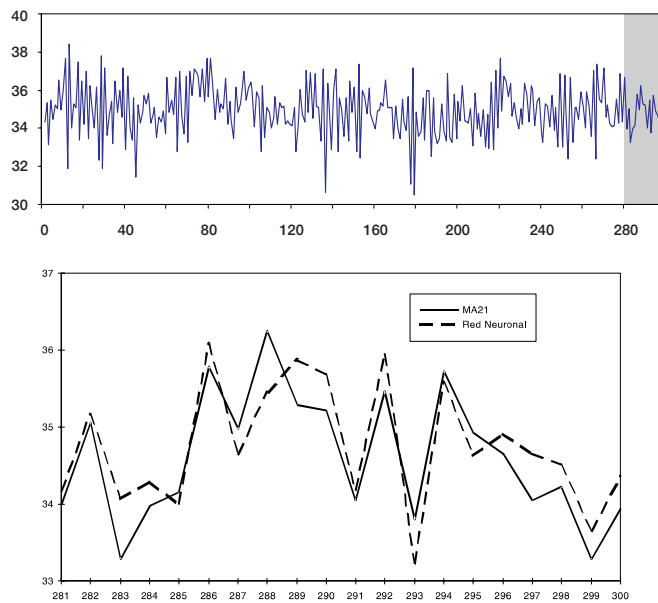


Figura 5.3. Serie MA21 (modelo MA(2) no invertible).

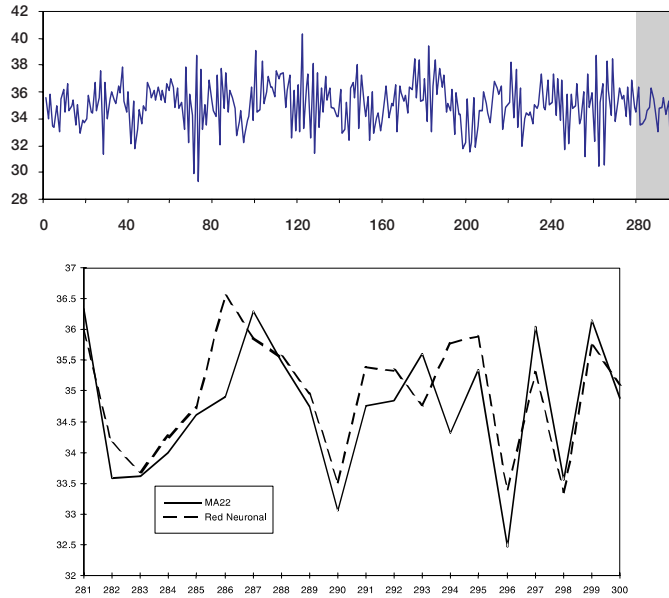


Figura 5.4. Serie MA22 (modelo MA(2) no invertible).

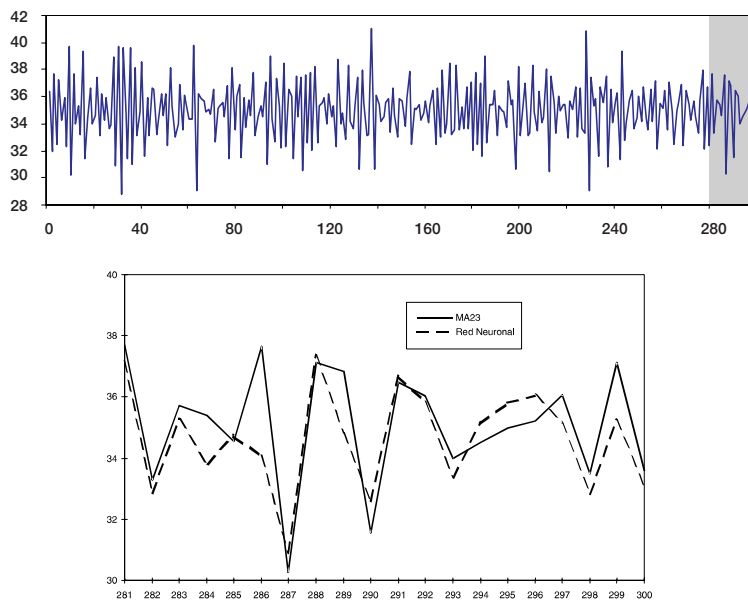


Figura 5.5. Serie MA23 (modelo MA(2) no invertible).

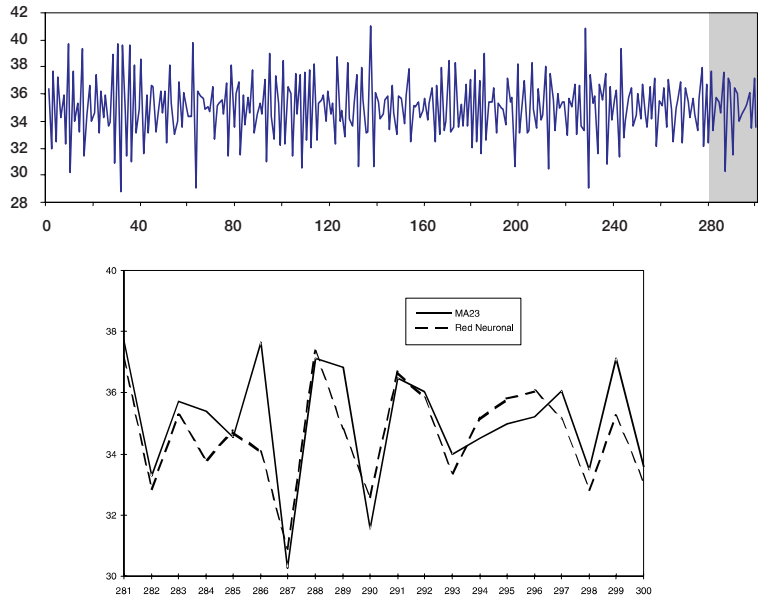


Figura 5.5. Serie MA23 (modelo MA(2) no invertible).

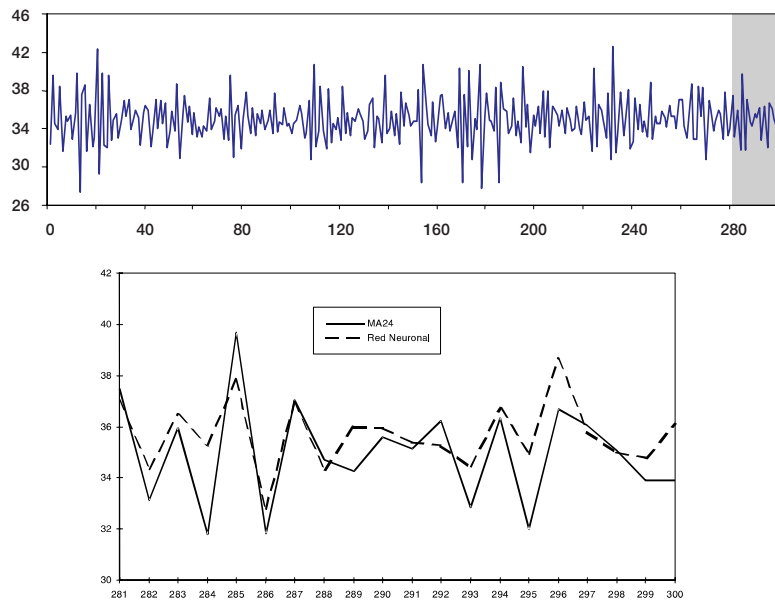


Figura 5.6. Serie MA24 (modelo MA(2) no invertible).

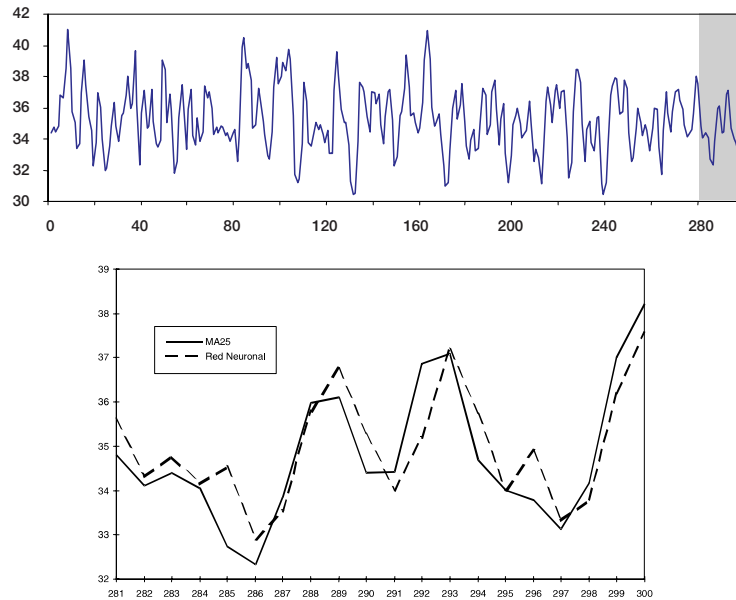


Figura 5.7. Serie MA25 (modelo MA(2) no invertible).

Tabla 5.2. Resultados del cálculo de previsiones.

Serie Temporal	Arquitectura y Configuración		MAD (1)	MSE (2)	MAPE (3)
AR1	MLP	14-7-1	0.074267	0.009043	2.59 %
AR21	MLP	8-3-1	0.686762	0.601951	1.30 %
AR22	MLP	6-4-1	519103.7	5.93E+11	28.84 %
AR23	MLP	10-6-1	9.98E+22	1.50E+46	28.62 %
AR24	MLP	13-8-1	0.359097	0.179917	0.58 %
AR25	MLP	8-4-1	0.982656	1.360244	13.94 %
AR26	MLP	6-4-1	1.38E+26	3.05E+52	17.58 %
MA20	ELM	16-9-1	0.565675	0.461447	1.63 %
MA21	MLP	11-7-1	0.378792	0.185006	1.09 %
MA22	ELM	8-5-1	0.531488	0.449266	1.52 %
MA23	ELM	10-6-1	0.894204	1.442150	2.56 %
MA24	ELM	10-6-1	1.120516	2.158522	3.21 %
MA25	MLP	12-7-1	0.617539	0.622427	1.77 %

(1) MAD (Mean Absolute Deviation).

(2) MSE (Mean Squared Error).

(3) MAPE (Mean Absolute Percentage Error).

En la tabla 5.2, se muestran las configuraciones de las redes seleccionadas para calcular las previsiones de todas estas series no estacionarias o no invertibles, y los estadísticos empleados para medir la calidad de las previsiones.

Es destacable el hecho de que en todos los modelos autorregresivos estudiados, la mejor arquitectura encontrada fue el MLP, mientras que en los modelos de media móvil, en la mayoría de los casos se obtuvieron mejores resultados con redes recurrentes de Elman. Esto confirma los estudios de Dorffner (1996), en los que se identifican los modelos *autorregresivos* de Box-Jenkins con el Perceptrón Multicapa y los modelos *media móvil* con la filosofía del funcionamiento de las redes parcialmente recurrentes (de Jordan o de Elman).

6. SISTEMA MULTIVARIANTE NO ESTACIONARIO

Un caso particularmente interesante es el de los sistemas multivariantes no estacionarios. A la dificultad propia de los sistemas multivariantes, se suma el hecho de que alguna o todas las series que componen el sistema, sean no estacionarias.

Presentamos como ejemplo el sistema denominado ARUMA33 (Huang and Anh, 1993), que se puede describir mediante la ecuación:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} B \\ B \end{bmatrix}^2 \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \varepsilon_{t1} \\ \varepsilon_{t2} \end{bmatrix}$$

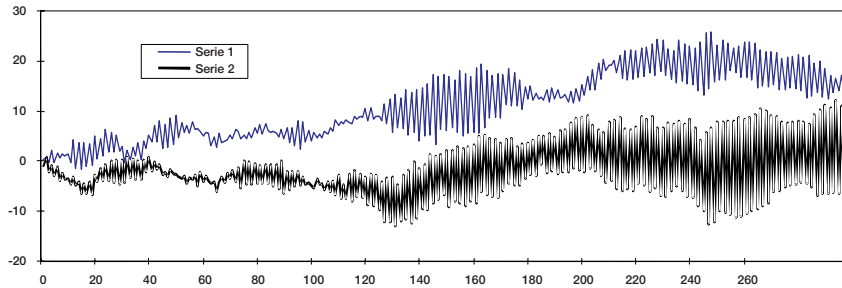


Figura 6.1. Sistema Multivariante ARUMA33.

En la figura 6.1, se han representado las dos series que componen el sistema multivariante y en la figura 6.2, las previsiones calculadas para cada una de ellas. En este caso se han utilizado dos redes neuronales: un perceptrón multicapa 12-8-1 para pronosticar la Serie 1, y otro perceptrón 12-9-1 para calcular las previsiones de la Serie 2. Igual que

en los casos anteriores, se han llevado a cabo distintas pruebas para encontrar el número adecuado de neuronas en la capa oculta de cada una de las redes.

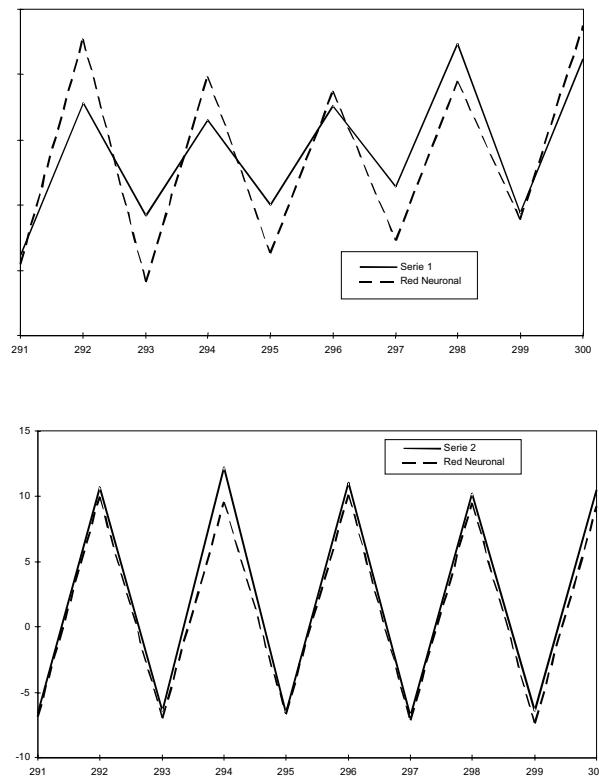


Figura 6.2. Previsiones para el Sistema Multivariante ARUMA33.

7. CONCLUSIONES

En este trabajo se ha demostrado el potencial de las Redes Neuronales Artificiales como herramienta para el cálculo de previsiones de series temporales.

En aquellos casos en los que aplicar la metodología clásica de Box-Jenkins es difícil o lleva a resultados inaceptables debido a las características de la serie temporal que se desea pronosticar, por ejemplo cuando se trata de series no estacionarias y/o no invertibles, es necesario realizar una serie de transformaciones de la serie temporal para que se asegure la estacionariedad y/o la invertibilidad.

Las Redes Neuronales gracias a su capacidad de «aprender» a partir de los ejemplos que se le proporcionan, sin hacer suposiciones a priori sobre los modelos y relaciones que subyacen en la serie, pueden ser aplicadas directamente, lo que facilita considerablemente el proceso de cálculo de previsiones.

Hemos querido demostrar este hecho, calculando previsiones de varias series temporales: 7 modelos autorregresivos no estacionarios, 6 modelos media móvil no invertibles, y un sistema multivariante no estacionario de 2 entradas y 2 salidas.

En el caso de las series univariantes, la arquitectura que se demostró más adecuada para los modelos autorregresivos fue el MLP, mientras que para los modelos de media móvil, los mejores resultados se obtuvieron con redes recurrentes de Elman. Los problemas de la aplicación de la metodología de Box-Jenkins, son aún mayores cuando se aplican al cálculo de previsiones de sistemas multivariantes, mientras que en el caso de las redes neuronales, su tratamiento no difiere del necesario para los sistemas univariantes, como hemos podido demostrar calculando pronósticos de un sistema multivariante no estacionario.

REFERENCES

- Huang, D. and Anh, V. V. (1993). «Estimation of the Non-Stationary Factor in AR-MA Models». *Journal of Time Series Analysis*, 14(1), 27-46.
- Box, G. E. and Jenkins, G. M. (1970). *Time Series Analysis*, Holden-Day, San Francisco.
- Debar, H. and Dorizzi, B. (1992). «An Application of a Recurrent Network to an Intrusion Detection System». *Proceedings of the IEEE International Joint Conference on Neural Networks*, Baltimore.
- Dorffner, G. (1996). «Neural Networks for Time Series Processing», *Neural Network World*, 6(4), 447-468.
- Dorffner, G.; Leitgeb, E. and Koller, H. (1994). «Towards Improving Exercise ECG for Detecting Ischemic Heart Disease with Recurrent and Feedforward Neural Nets». *Neural Networks for Signal Processing IV*, Vlontzos, J. et al. (eds.), IEEE, New York, 499-508.
- Elman, J. L. (1990). «Finding structure in time». *Cognitive Science*, 14, 179-211.
- Freeman, J. A. and Skapura, D. M. (1991). *Neural Networks: Algorithms, Applications, and Programming Techniques*. Addison-Wesley Publishing Company, Inc., Reading, MA.
- Gordon, A.; Steele, J. P. H. and Rossmiller, K. (1991). «Predicting Trajectories Using Recurrent Neural Networks». *Intelligent Engineering Systems through Artificial Neural Networks*, Dagli, C.H. et al. (eds.), ASME Press, New York, 365-370.

- Huang, D. (1990). «Levinson-type recursion for least squares estimation». *Journal of Time Series Analysis*, 11, 295-315.
- Jordan, M. I. (1986). «Attractor dynamics and parallelism in a connectionist sequential machine». *Proceedings of the Eight Annual Conference of the Cognitive Science Society*, 531-546, Erlbaum.
- Kamijo, K. and Tanigawa, T. (1993). «Stock Price Pattern Recognition: A Recurrent Neural Network Approach». *Neural Networks in Finance and Investing*, Trippi, R. R. *et al.* (eds.), 357-370.
- Lee, C. H. and Park, K. C. (1992). «Prediction of Monthly Transition of the Composition Stock Price Index using Recurrent Back-Propagation». *Artificial Neural Networks 2*, Aleksander, I. *et al.* (eds.), 1629-1632.
- Lippmann, R. P. (1987). «An Introduction to Computing with Neural Networks». *IEEE ASSP Magazine*, 3(4), 4-22.
- Rumelhart, D. E. and McClelland, J. (1986). *Parallel Distributed Processing*. MIT Press, Cambridge.
- Rumelhart, D. E.; Hinton, G. E. and Williams, R. J. (1986). «Learning representations by back-propagating errors». *Nature*, 323, 533-536.
- SNNS (1998). *Stuttgart Neural Network Simulator. User Manual, Version 4.2*. University of Stuttgart.
- Zhang, G.; Patuwo, B. E. and Hu, M. Y. (1998). «Forecasting with artificial neural networks: The state of the art». *International Journal of Forecasting*, 14, 35-62.

ENGLISH SUMMARY

THE APPLICATION OF ARTIFICIAL NEURAL NETWORKS TO FORECASTING NON-STATIONARY OR NON-INVERTIBLE TIME SERIES

R. PINO*

D. DE LA FUENTE

J. PARREÑO

P. PRIORE

Universidad de Oviedo

There has recently been noticeably increased interest in the application of Artificial Neural Networks to forecasting times series, and attempts have been made to exploit the undoubted advantages of these tools. This paper deals with non-stationary or noninvertible times series that are problematic if forecasting uses the Box-Jenkins' ARIMA methodology. The advantages of applying neural networks can be appreciated more clearly in the case of multivariate non-stationary system forecasting.

Keywords:

AMS Classification (MSC 2000): 37M10 Time Series Analysis
62M10 Time Series
62M45 Neural Networks
82C32 Neural Networks

*Dpto. Admon. de Empresas y Contabilidad. Escuela Politécnica de Ingenieros de Gijón. Campus de Viesques, s/n. 33204 GIJÓN (Spain). E-Mail: pino@etsiig.uniovi.es.

–Received May 2001.

–Accepted October 2002.

1. INTRODUCTION

It is well known that a time series has to meet the stationarity and invertibility conditions in order to be correctly modelled by an ARIMA model. Autoregressive (AR) models meet these conditions if all the roots of the characteristic polynomial are outside the unit circle; the same condition also ensures the invertibility of Moving Average (MA) models. For autoregressive and moving average (ARMA) models, several conditions exist that guarantee stationarity and invertibility for certain locations of the roots of the characteristic polynomial of AR and MA factors.

If these conditions are not met, Box-Jenkins methodology would lead to forecasts beyond reasonable limits, so that it would be necessary to undertake some prior transformations to the time series before estimating the model and then making the forecast.

One of the characteristics of Artificial Neural Nets (ANNs) resides in their ability to «learn» from the examples they are provided with, without making a priori suppositions about the models and underlying correlations in a given time series. This fact makes possible to calculate forecasts for any time series without having to previously test the stationarity and invertibility.

The aim of this job is to demonstrate the affirmations made in the previous paragraph by using ANNs to calculate forecasts for several non-stationary or non-invertible time series.

2. ARTIFICIAL NEURAL NETS

This section will briefly describe the structure of Multilayer Perceptron (MLP) and Recurrent Neural Nets (RNNs).

MLP, also known as Backpropagation Net (BPN) is one of the most used ANN models as pattern classifier and function approximator. It belongs to the category of feedforward nets, and is structured in layers where the information flows from the input layer, through the hidden layers, to the output layer. The information processing is responsibility of the hidden layers.

The most commonly training algorithm for MLP is the Backpropagation (BP), an algorithm that is widely used because of its simplicity and efficiency.

The second type of net used in this study are Recurrent Neural Nets (RNN). These nets have an architecture based on MLP, but with certain modifications, as the presence of a contextual layer. The ability of RNNs for detecting the time evolution of a variable during consecutive time periods makes them very interesting and adequate for forecasting.

3. RESULTS

Forecasts were calculated for several time series: seven non-stationary AR models, six non-invertible MA models and a 2-input 2-output non-stationary multivariate system. For all these series, Box-Jenkins methodology showed little forecasting accuracy.

For the univariate cases, MLP proved to be the more adequate architecture for AR models, whilst best results for MA models were obtained with RNNs. For the multivariate case, the best results were provided by MLP.