A Compounded Multi-resolution-Artificial Neural Network Method for the Prediction of Time Series with Complex Dynamics

Livio Fenga

Abstract Time series realizations of stochastic process exhibiting complex dynamics are dealt with. They can be affected by a number of phenomena, like asymmetric cycles, irregular spikes, low signal-to-noise ratio, chaos, and various sources of turbulences. Linear models are not designed to perform optimally in such a context, therefore a mixed self-tuning prediction method—able to account for complicated patterns—is proposed. It is a two-stage approach, exploiting the multi-resolution capabilities delivered by Wavelet theory in conjunction with artificial neural networks. Its out-of-sample forecast performances are evaluated through an empirical study, carried out on macroeconomic time series.

Keywords Complex dynamics • Feed-forward artificial neural networks • Maximum overlapping discrete wavelet transform • Time series forecast

1 Introduction

One of the most effective and well-established practical uses of time series analysis is related to the prediction of future values using its past information [1]. However, much of the related statistical analysis is done under linear assumptions which—outside trivial cases and ad hoc lab—controlled experiments—hardly ever do possess features compatible with real-world *DGP*s. The proposed forecast procedure has been designed to account for complex, possibly non-linear dynamics one may encounter in practice and combines the wavelet multi-resolution decomposition approach, with a non-standard, highly computer intensive statistical method: artificial neural networks (*ANN*). The acronym chosen for i—i.e. *MUNI*, short for Multi-resolution Neural Intelligence—reflects both these aspects. In more details, *MUNI* procedure is based on the reconstruction of the original time series after its decomposition, performed through an algorithm based on the inverse of a wavelet transform, called Multi-resolution Approximation (*MRA*) [2–4]. Practically, the

L. Fenga (✉)
UCSD, University of California San Diego, La Jolla, CA, USA
e-mail: lfenga@math.ucsd.edu

I. Rojas, H. Pomares (eds.), *Time Series Analysis and Forecasting*, Contributions to Statistics, DOI 10.1007/978-3-319-28725-6_27

original time series is decomposed into more elementary components (first step), each of them representing an input variable for the predictive model, and as such individually predicted and finally combined through the inverse *MRA* procedure (second step). In charge of generating the predictions is the time domain—Artificial Intelligence (*AI*)—part of the method which exploits an algorithm belonging to the class of parallel distributed processing, i.e., *ANN* [5, 6], with an input structure of the type autoregressive.

1.1 Signal Decomposing and Prediction Procedures

In what follows the time series (signal) of interest is assumed to be real-valued, uniformly sampled of finite length T, i.e.: $x_t := \left\{(x_t)_{t\in\mathbb{Z}^+}^T\right\}$. *MUNI* has been implemented with a wavelet [7–9] signal-coefficient transformation procedure of the type Maximum Overlapping Discrete Wavelet Transform (*MODWT*) [10], which is a filtering approach aimed at modifying the observed series $\{x\}_{t\in Z^+}$, by artificially introducing an extension of it, so that the unobserved samples $\{x\}_{t\in Z^-}$ are assigned the observed values $X_{T-1}, X_{T-2}, \ldots, X_0$. This method considers the series as it were periodic and is known as using circular boundary conditions, where wavelet and scale coefficients are respectively given by:

$$d_{j,t} = \frac{1}{2^{j/2}} \sum_{l=0}^{L_j-1} \tilde{h}_{j,l}, X_{t-l \bmod N}, \qquad S_{J,t} = \frac{1}{2^{j/2}} \sum_{l=0}^{L_j-1} \tilde{g}_{j,l}, X_{t-l \bmod N},$$

with $\left\{\tilde{h}_{j,l}\right\}$ and $\left\{\tilde{g}_{j,l}\right\}$ denoting the length L, level j, wavelet, and scaling filters, obtained by rescaling their Discrete Wavelet Transform counterparts, i.e., $\left\{h_{j,l}\right\}$ and $\left\{g_{j,l}\right\}$, as follows: $\tilde{h}_{j,l} = \frac{h_{j,l}}{2^{j/2}}$ and $\tilde{g}_{j,l} = \frac{g_{j,l}}{2^{j/2}}$. Here, the sequences of coefficients $\left\{h_{j,l}\right\}$ and $\left\{g_{j,l}\right\}$ are approximate filters: the former of the type band-pass, with nominal pass-band $f \in [\frac{1}{4\,\mu_j}, \frac{1}{2}\mu_j]$, and the latter of the type low-pass, with a nominal pass-band $f \in [0, \frac{1}{4}\mu_j]$, with μ_j denoting the scale. Considering all the $J = J^{\max}$ sustainable scales, *MRA* wavelet representation of x_t, in the $L^2(R)$ space, can be expressed as follows:

$$\begin{aligned} x(t) = &\sum_k s_{J,k}\phi_{J,k}(t) + \sum_k d_{J,k}\psi_{J,k}(t) + \sum_k d_{J-1,k}\psi_{J-1,k}(t) + \cdots \\ &+ \sum_k d_{j,k}\psi_{j,k}(t) \cdots + \sum_k d_{1,k}\psi_{1,k}(t), \end{aligned} \tag{1}$$

with k taking integer values from 1 to the length of the vector of wavelet coefficients related to the component j and ψ and ϕ, respectively, the father and mother wavelets (see, for example, [11, 12]). Assuming that a number $J_0 \leq J^{\max}$ of scales is selected, *MRA* is expressed as $x_t = \sum_{j=1}^{J_0} D_j + S_{J_0}$, with $D_j = \sum_k d_{J,k}\psi_{J,k}(t)$ and $S_j = \sum_k s_{J,k}\phi_{J,k}(t)$; $j = 1, 2, \ldots, J$. Each sequence of coefficients d_j, (in signal processing called crystal), represents the original signal at a given resolution level, so that the *MRA* conducted at a given level j $(j = 1, 2, \ldots, J)$, delivers the coefficients set D_j, which reflects signal local variations at the detailing level j, and the set S_{J_0}, accounting for the long run variations. By adding more levels $\{d_j;\ j = 1, 2, \ldots, J^{\max}\}$, finer levels js are involved in the reconstruction of the original signal and the approximation becomes closer and closer, until the loss of information becomes negligible. The forecasted values are generated by aggregation of the predictions singularly obtained by each of the wavelet components, once they are transformed via the Inverse *MODWT* algorithm, i.e.: $\hat{x}_t(h) = \sum_{j=1}^{J_0} \hat{D}_j^{\text{inv}}(h) + \hat{S}_{J_0}^{\text{inv}}(h)$, where D and S are as above defined and the superscript inv indicates the inverse *MODWT* transform. In total, four are the choices required for a proper implementation of *MODWT*, they are boundary conditions, type of wavelet filter, its width parameter L, and number of decomposition levels. Regarding the first choice, *MUNI* has been implemented with periodic boundary conditions. However, alternatives can be evaluated on the basis of the characteristics of the time series and/or as a part of a preliminary investigation. The choices related to the type of wavelet function and its length L are generally hard to automatize, therefore their inclusion in *MUNI* has not been pursued. More simply, it has been implemented with the fourth order Daubechies least asymmetric wavelet filter (known also as symmlets) [8] of length $L = 8$, usually denoted $LA(8)$. Regarding the forecasting method, *MUNI* uses a neural topology belonging to the family of multilayer perceptron [13, 14], of the type feed-forward (*FFWD-ANN*) [15]. This is a popular choice in computational intelligence for its ability to perform in virtually any functional mapping problem including autoregressive structures. This network represents the non-linear function mapping from past observations $\{x_{t-\tau};\ (\tau = 1, 2, \ldots, T-1)\}$ to future values $\{x_h;\ (h = T, T+1, \ldots)\}$, i.e.: $x_t = \sigma_{nn}\left(x_{t-1}, x_{t-2}, \ldots, x_{t-p}, \mathbf{w}\right) + u_t$, with p the maximum autoregressive lag, $\sigma(.)$ the activation function defined over the inputs and $\mathbf{w}$ the network parameters and u_t the error term. In practice, the input–output relationship is learnt by linking, via acyclic connections, the output x_t to its lagged values, constituting the network input, through a set of layers. While the latter has usually a very low cardinality (often 1 or 2), the input set is critical—being the inclusion of not-significant lags and/or the exclusion of significant ones able to affect the quality of the outcomes.

1.1.1 The Learning Algorithm and the Regularization Parameter η

MUNI envisions the time series at hand split in three different, non overlapping parts, serving respectively as training, test, and validation sets. The training set is the sequence $\{(\mathbf{x}_1, \mathbf{q}_1), \ldots, (\mathbf{x}_p, \mathbf{q}_p)\}$, in the form of p ordered pairs of n- and m-dimensional vectors, were $\mathbf{q}_i$ denotes the target value and $\mathbf{x}_i$ the matrix of the delayed time series values. The network, usually initialized with random weights, is presented an input pattern and an output, say $\mathbf{o}_i$, is generated as a result. Being in general $\mathbf{o}_i \neq \mathbf{q}_i$, the learning algorithm tries to find the optimal weights vector minimizing the error function in the $\mathbf{w}$-space, that is: $\mathbf{o}_p = f_{nn}(\mathbf{w}, \mathbf{x}_p)$, where the weight vector $\mathbf{w}$ refers, respectively, to the p_i output and the p_i input and f_{nn} the activation function. Denoting here the training set with T_r and with P_{tr} the number of pairs, the average error E committed by the network can be expressed as: $\tilde{E}(\mathbf{w}) = E(\mathbf{w}) + \frac{1}{2}\eta \sum_i \mathbf{w}_i^2$, where η is a constraint term aimed at penalizing model weights and thus limiting the probability of over-fitting.

1.1.2 Intelligent Network Parameters Optimization

In this section, *MUNI*'s *AI*-driven part is illustrated and some notation introduced. In essence, it is a multi-grid searching system for the estimation of an optimal network vector of parameters under a suitable loss function, i.e., the root mean square error (*RMSE*), expressed as

$$\mathfrak{B}(x_i, \hat{x}_i) = \left[T^{-1} \sum_{i}^{T} |e_i|^2 \right]^{\frac{1}{2}}, \tag{2}$$

with x_i denoting the observed value, e the difference between it and its prediction $\hat{x}_i$, and T the sample size. The parameters subjected to neural-driven search, listed in Table 1 along with the symbols used to denote each of themare stored in the vector $\boldsymbol{\omega}$, i.e.: $\boldsymbol{\omega} \equiv (\beta, \rho, \alpha, \eta, \lambda, \nu)$. Each of them is associated a grid, whose arbitrarily chosen values are all in $\mathbb{Z}^+$. Consistently with the list of parameters of Table 1, the set of these grids, is formalized as follows: $\boldsymbol{\Gamma} = (\{\Gamma_\beta\}, \{\Gamma_\rho\}, \{\Gamma_\alpha\}, \{\Gamma_\eta\}, \{\Gamma_\lambda\}, \{\Gamma_\nu\})$, where each subset $\{\Gamma_{(\cdot)}\}$ has cardinality respectively equals to $\tilde{\beta}$, $\tilde{\rho}$, $\tilde{\alpha}$, $\tilde{\eta}$,

Table 1 *MUNI*'s parameters and related notation

Symbol	Parameter
β	Number of the sets of decomposition levels
ρ	Rescaling factor
α	Number of iterations
η	Decay ratio
λ	Number of input neurons (lag-set)
ν	Number of hidden layer neurons

$\tilde{\lambda}\tilde{\nu}$.[1] The wavelet-based *MRA* procedure is applied $\tilde{\beta}$ times, so that the time series of interest is broken down into $\tilde{\beta}$ different sets, each containing different numbers of crystals, in turn contained by a set denominated A, i.e.: $\{A_{\beta_w};\ \ w = 1, 2, \ldots, \tilde{\beta}\} \subset A$. Here, each of the A's encompasses a number of decomposition levels ranging from a minimum and a maximum, respectively, denoted by $J^{\min}$ and $J^{\max}$, therefore, for the generic set $A_{\beta_w} \subset A$, it will be:

$$A_{\beta_w} = \left\{J^{\min} \leq k, k+1, k+2, \ldots, K \leq J^{\max};\ \ J^{\min} > 1\right\}. \tag{3}$$

Assuming a resolution set A_{β_0} and a resolution level $k_0 \subset A_{\beta_0}$, the related crystal, denoted by $\mathcal{X}_{k_0,\beta_0}$, is processed by the network $\mathfrak{N}_1$, which is parametrized by the vector of parameters $\omega_1^{(k_0,A_{\beta_0})} \equiv (\rho^{(k_0,A_{\beta_0})}, \alpha^{(k_0,A_{\beta_0})}, \eta^{(k_0,A_{\beta_0})}, \lambda^{(k_0,A_{\beta_0})}, \nu^{(k_0,A_{\beta_0})})$. Once trained, the network $\mathfrak{N}_1$, denoted by $\tilde{\mathfrak{C}_1}^{k_0,A_{\beta_0}}$, is employed to generate H-step ahead predictions. *MUNI* chooses the best parameter vector, i.e., ${}^*\boldsymbol{\omega}^{(k_0,A_{\beta_0})}$ for $\mathcal{X}_{k_0,\beta_0}$, according to the minimization of a set of cost functions of the form $\mathfrak{B}(\mathcal{X}_{k_0,\beta_0}, \hat{\mathcal{X}}_{k_0,\beta_0})$ iteratively computed on the predicted values $\hat{\mathcal{X}}_{k_0,\beta_0}$ in the validation set. These predictions are generated by a set of networks parametrized and trained according to the set of k-tuples (with k the length of ω) induced by the set of the Cartesian relations on Γ. Denoting the former by $\tilde{\mathfrak{C}}^{k_0,A_{\beta_0}}$ and by $\mathbf{P}$ the latter, it will be:

$${}^*\boldsymbol{\omega}^{(k_0,A_{\beta_0})} = \arg\min_{\boldsymbol{P}}\ \mathfrak{B}(\hat{\mathcal{X}}_{k_0,A_{\beta_0}}(\boldsymbol{P}), \mathcal{X}_{k_0,A_{\beta_0}}). \tag{4}$$

The set of all the trained networks attempted at the resolution level A_{β_0} (i.e., encompassing all the crystals in A_{β_0}), is denoted by $\tilde{\mathfrak{C}}^{A_{\beta_0}}$, whereas the set of networks trained in the whole exercise (i.e., for all the A's), by $\tilde{\mathfrak{C}}^{A}$. The networks $\tilde{\mathfrak{C}}^{A_{\beta_0}}$, parametrized with the optimal vector $({}^*\boldsymbol{\omega}_{J^{\min}}, \ldots, {}^*\boldsymbol{\omega}_{J^{\max}}) \equiv {}^*\boldsymbol{\Omega}_{\beta_0}$, which is obtained by applying (4) to each crystal, are used to generate predicted values at each resolution level independently. These predictions are combined via Inverse-*MODWT* and evaluated in terms of the loss function $\mathfrak{B}$, computed on the validation set of the original time series. By repeating the above steps for the remaining sets, i.e., $\{A_{\beta_w};\ \ w = 1, 2, \ldots, \tilde{\beta} - 1\} \subset A$, $\tilde{\beta}$ optimal sets of networks ${}^*\tilde{\mathfrak{C}}^{A}$, each parametrized by optimal vectors of parameters $\{{}^*\boldsymbol{\Omega}_w;\ \ w = 1, 2, \ldots, \tilde{\beta}\}$ are obtained. Each set of networks in ${}^*\tilde{\mathfrak{C}}^{A}$ is used to generate one vector of predictions

[1] For example, for the grid Γ_λ we have $\Gamma_\lambda = \Gamma_\lambda^{(1)}, \Gamma_\lambda^{(2)}, \ldots, \Gamma_\lambda^{(\tilde{\lambda})}$, with the generic element $\Gamma_\lambda^{(j)}$ denoting one of the values chosen for the number of the input neurons.

Table 2 *MUNI* procedure: a priori activities and choices

Decomposition unit	*AI* unit
Data exploration	
Data pre-processing	
Waveform and its length	Network topology
Wavelet transform method	Number of hidden layers
Boundary conditions	Sets of grids
Reconstruction algorithm	Squashing function
	Prediction horizon

for x_t in the validation set (by combination of the multi-resolutions predictions via *MODWT*), so that, by iteratively applying (2) to each of them, a vector containing $\tilde{\beta}$ values of the loss function, say $\mathfrak{L}_w$, is generated. Finally, the set of networks in a resolution set say ${}^*A$, whose parametrizations minimize $\mathfrak{L}_w$, are the winners, i.e.: ${}^*\boldsymbol{\Omega}^A = \underset{({}^*\boldsymbol{\Omega}_w)}{\arg\min}\ (\mathfrak{L})$.

1.1.3 Human-Driven Decisions

Being *MUNI* a partially self-regulating method, while it embodies automatic estimation procedures for a number of parameters, it requires a set of preliminary, human-driven choices, involving both the decomposition and the *AI* parts, as summarized in Table 2. Here, the first two entries are in common in that refer to activities which are not unit specific.

1.1.4 The Algorithm

MUNI procedure is now detailed in a step-by-step fashion.

1. Let x_t be the time series of interest [as defined in of interest (Sect. 1.1)], split in three disjoint segments: training set $\left\{x^{\mathrm{Tr}}\right\}_t$; $t = 1, 2, \ldots, T-(S+V+1)$, validation set, $\left\{x^U\right\}_t$ $t = T-(S+V+1), \ldots, (T-V)$ and test set, $\left\{x^S\right\}_t$ $t = T-(V+1), \ldots, T$), where with V and S, respectively, the length of validation and test set are denoted;
2. *MODWT* is applied $\tilde{\beta}$ times to x_t^{Tr}, and the related sets of crystals are stored in $\tilde{\beta}$ different sets, which in turn are stored in the matrix $\mathfrak{D}_{j,w}$, of dimension $(J^{\mathrm{MAX}} \times \tilde{\beta})$ of the form[2]:

[2]The upper row containing the symbols for the A's has been added for clarity.

$$\mathfrak{D}_{j,w} = \begin{array}{c} \begin{matrix} A_{\beta_1} & A_{\beta_2} & & A_{\beta_w} & & A_{\tilde{\beta}} \end{matrix} \\ \begin{bmatrix} \mathcal{X}_{J^{\text{MIN}},1} & \mathcal{X}_{J^{\text{MIN}},2} & \cdots & \mathcal{X}_{J^{\text{MIN}},w} & \cdots & \mathcal{X}_{J^{\text{MIN}},\tilde{\beta}} \\ \mathcal{X}_{J^{\text{MIN}}+1,1} & \mathcal{X}_{J^{\text{MIN}}+1,2} & \cdots & \mathcal{X}_{J^{\text{MIN}}+1,w} & \cdots & \mathcal{X}_{J^{\text{MIN}}+1,\tilde{\beta}} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \mathcal{X}_{k,1} & \mathcal{X}_{k,2} & \cdots & \mathcal{X}_{k,w} & \cdots & \mathcal{X}_{k,\tilde{\beta}} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \mathcal{X}_{J^{\text{MAX}},1} & \vdots & \cdots & \vdots & \cdots & \vdots \\ \emptyset & \vdots & \vdots & \vdots & \cdots & \vdots \\ \emptyset & \mathcal{X}_{J^{\text{MAX}},2} & \cdots & \vdots & \cdots & \vdots \\ \emptyset & \emptyset & \vdots & \vdots & \cdots & \vdots \\ \emptyset & \emptyset & \cdots & \mathcal{X}_{J^{\text{MAX}},w} & \cdots & \vdots \\ \emptyset & \emptyset & \vdots & \emptyset & \cdots & \vdots \\ \emptyset & \emptyset & \cdots & \emptyset & \cdots & \mathcal{X}_{J^{\text{MAX}},\tilde{\beta}} \end{bmatrix} \end{array}.$$

Here, the generic column β_w, represents the set of resolution levels generated by a given *MRA* procedure, so that its generic element $\mathcal{X}_{k,w}$ is the crystal obtained at a given decomposition level k belonging to the set of crystals A_{β_w}. For each column vector β_w, a minimum and a maximum decomposition level (3), $J^{\min}$ and $J^{\max}$, is arbitrarily chosen;

3. the set $\mathbf{P}$ of the parametrizations of interest is built. It is the set of all the Cartesian relations $\mathbf{P} \equiv \{\Gamma_\rho \times \Gamma_\alpha \times \Gamma_\eta \times \Gamma_\lambda \times \Gamma_\nu\}$ whose cardinality, expressed through the symbol $|\cdot|$, is denoted by $|\mathbf{P}|$;
4. an arbitrary set of decomposition levels, say $A_0 \subset A$, is selected (the symbol β is suppressed for an easier readability);
5. an arbitrary crystal, say $\mathcal{X}_{k_0,A_0} \subset A_0$, is extracted;
6. the parameter vector $\boldsymbol{\omega}$ is set to an (arbitrary) initial status $\boldsymbol{\omega}_1 \equiv \mathbf{P}_1 \equiv \left[\, \Gamma_\alpha^{(1)}, \Gamma_\rho^{(1)}, \Gamma_\eta^{(1)}, \Gamma_\lambda^{(1)}, \Gamma_\nu^{(1)} \,\right]$;
   (a) $\mathcal{X}_{k_0,A_0}$ is submitted to and processed by a single hidden layer *ANN* of the form

$$\begin{cases} \mathfrak{N}_{(1)} = \sigma\left(\mathcal{X}_{k_0,A_0}, \mathbf{w}_{(1)}\right) \\ \mathbf{w}_{(1)} = \sigma\left(\boldsymbol{\omega}_1^{k_0,A_0}\right), \end{cases}$$

   with σ being the sigmoid activation function and $\mathbf{w}$ the network weights evaluated for a given configuration of the parameter vector, i.e., $\boldsymbol{\omega}_1$.
   (b) network $\mathfrak{N}_1$ is trained and the network $\tilde{\mathfrak{C}_1}^{k_0,A_0}$, obtained as a result, is employed to generate H-step ahead predictions for the validation set x^U. These predictions are stored in the vector $\mathfrak{P}_1^{k_0,A_0}$;

(c) steps 6a–6d are repeated for each of the remaining $(|\mathbf{P}|-1)$ elements of $\mathbf{P}$. The matrix $\mathfrak{P}_m^{k_0,A_0}$ of dimension $\left(U \times |\mathbf{P}|-1\right)$, containing the related predictions (for the crystal $\mathcal{X}_{k_0,A_0}$) is generated by the trained networks $\tilde{\mathfrak{C}}_{2,\ldots,|\mathbf{P}|}^{k_0,A_0}$;

(d) the matrix ${}_{\text{full}}\mathfrak{P}^{k_0,A_0}$ of dimensions $\left(U \times |\mathbf{P}|\right)$ containing all the predictions for the crystal $\mathcal{X}_{k_0,A_0}$ is generated, i.e.,

$$ {}_{\text{full}}\mathfrak{P}^{k_0,A_0} = \mathfrak{P}_1^{k_0,A_0} \quad \cup \quad \mathfrak{P}_m^{k_0,A_0}; $$

(e) steps 6a–6d are repeated for each of the remaining crystals in A_0, i.e.,

$$ \left\{\mathcal{X}_{k_i,A_0}; \quad i = J^{\min}, J^{\min+1}, \ldots, (J^{\max}-1)\right\} \subset A_0 \subset A, $$

so that $i = 1, 2, \ldots, (J^{\text{MAX}}-1)$ prediction matrices $\mathfrak{P}^{k_i,A_0}$ are generated;

(f) the $\left((J^{\text{MAX}} - J^{\min}+1) \times |\mathbf{P}|\right)$ dimension matrix $\hat{\mathfrak{J}}_{A_0} \equiv_{\text{full}} \mathfrak{P}^{k_0,A_0} \cup \mathfrak{P}^{k_i,A_0}; \quad i = J^{\min}, J^{\min+1}, \ldots, (J^{\max}-1)$, containing all the predictions for the validation set $\left\{\mathcal{X}^U\right\}_t$, of all the crystals in A_0, is generated,[3] i.e.:

$$ \hat{\mathfrak{J}}_{A_0} = \begin{bmatrix} \hat{\mathcal{X}}_{J^{\text{MIN}},\omega_1} & \hat{\mathcal{X}}_{J^{\text{MIN}},\omega_2} & \cdots & \hat{\mathcal{X}}_{J^{\text{MIN}},\omega_k} & \cdots & \hat{\mathcal{X}}_{J^{\text{MIN}},\omega_{|\mathbf{P}|}} \\ \hat{\mathcal{X}}_{J^{\text{MIN}}+1,\omega_1} & \hat{\mathcal{X}}_{J^{\text{MIN}}+1,\omega_2} & \cdots & \hat{\mathcal{X}}_{J^{\text{MIN}}+1,\omega_k} & \cdots & \hat{\mathcal{X}}_{J^{\text{MIN}}+1,\omega_{|\mathbf{P}|}} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \hat{\mathcal{X}}_{k,\omega_1} & \hat{\mathcal{X}}_{k,\omega_2} & \cdots & \hat{\mathcal{X}}_{k,\omega_k} & \cdots & \hat{\mathcal{X}}_{k,\omega_{|\mathbf{P}|}} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ \hat{\mathcal{X}}_{J^{\text{MAX}},\omega_1} & \hat{\mathcal{X}}_{J^{\text{MAX}},\omega_2} & \cdots & \hat{\mathcal{X}}_{J^{\text{MAX}},\omega_k} & \cdots & \hat{\mathcal{X}}_{J^{\text{MAX}},\omega_{|\mathbf{P}|}} \end{bmatrix}; $$

7. loss function minimization in the validation set is used to build the set of winner *ANN*s for each of the crystals in A_0, i.e., $\mathfrak{C}_*^{A_0} \equiv \left\{{}^{J^{\min}}\mathfrak{C}_*^{A_0}, \ldots, {}^{J^{\max}}\mathfrak{C}_*^{A_0}\right\}$. For example, for the generic crystal k, the related optimal network is selected according to: ${}^{k}\mathfrak{C}_*^{A_0} = \arg\min_{\mathbf{P}} \mathfrak{B}(\mathcal{X}_k^U, \hat{\mathcal{X}}_k^U(\mathbf{P}))$;
8. $\mathfrak{C}_*^{A_0}$ is employed to generate the matrix $\hat{\mathcal{X}}_*^{A_0}$ of the optimal predictions for the validation set of each resolution level in A_0, i.e.,
$\hat{\mathcal{X}}_*^{A_0} \equiv \left[{}^{J^{\min}}\hat{\mathcal{X}}^U, \ldots\ldots, {}^{J^{\max}}\hat{\mathcal{X}}^U \right]'$;
9. by applying inverse *MODWT* to $\hat{\mathcal{X}}_*^{A_0}$, the series $\left\{x^U\right\}_t$ is reconstructed, i.e., $\mathfrak{Inv}(\hat{\mathcal{X}}_*^{A_0}) = \left\{\hat{x}_{A_0}^U\right\}_t$, so that the related loss function $\mathfrak{B}(x_t^U, \hat{x}_t^{(U)})$ is computed and its value stored in the vector $\boldsymbol{\Delta}$ whose length is $(J^{\text{MAX}} - J^{\min}+1)$;

[3] In order to save space, the empty set symbol $\emptyset$ is omitted.

10. steps 4–9 are repeated for the remaining sets of resolutions $A_1, \ldots A_w, \ldots, A_{\tilde{w}-1}$, so that all the $\tilde{w}$ error function minima are stored in the vector $\boldsymbol{\Delta}$;
11. the network set $\mathfrak{C}^*$ generating the estimation of the crystals minimizer of $\boldsymbol{\Delta}$ over all the network configurations $\mathfrak{C}$, is the final winner, i.e., $\mathfrak{C}^* = \arg\min_{\mathfrak{C}_*^A} \Delta(\mathfrak{C})$;
12. final performances assessments are obtained by using $\mathfrak{C}^*$ on the test set x_t^S.

2 Empirical Analysis

In this section, the outcomes of an empirical study conducted on four macroeconomic time series—i.e., Japan/USA Exchange rate, USA Civilian Unemployment rate (un-transformed and differenced data), Italian industrial production Index, respectively denoted *TS*1, *TS*2, *TS*3, *TS*4—are presented. These series (detailed in Tables 3 and 5) along with their empirical autocorrelation functions (*EACF*) [16], depicted respectively in Figs. 1 and 2, have been considered as they differ substantially for the type of phenomenon measured other than for their own inherent characteristics as time span, probabilistic structure, seasonality, and frequency components. In particular, *TS*2 and *TS*3, refer to the same variable (US civilian unemployment rate), and are included in the empirical analysis to emphasize *MUNI*'s capabilities to yield comparable results when applied to both the original and transformed data (and thus to simulate the case of a not pre-processed input series). As expected, the two series exhibit a different pattern: the un-transformed one (*TS*2), in fact, shows an ill behavior, in terms of both seasonal components and non-stationarity, in comparison with its differenced (the difference order is 1 and 12) counterpart *TS*3. On the other hand, *TS*1–2

Table 3 Specification of the time series employed in the empirical analysis

Name	Variable	Source	Period	Span	Transform
*TS*1	Japan/USA Exchange rate	Board of Governors of the Federal Reserve System	Jan. 1992–Jan. 2015	277	No
*TS*2	Civilian unemployment rate	US Bureau of Labor Statistics (Household survey)	Jan. 1948–Jan. 2015	805	No
*TS*3	Civilian unemployment rate	US Bureau of Labor Statistics (Household survey)	Jan. 1948–Jan. 2015	792	Diff(1,12)
*TS*4	Italian industrial production Index	Italian National Institute of Statistics	Jan. 1981–Jun. 2014	402	No

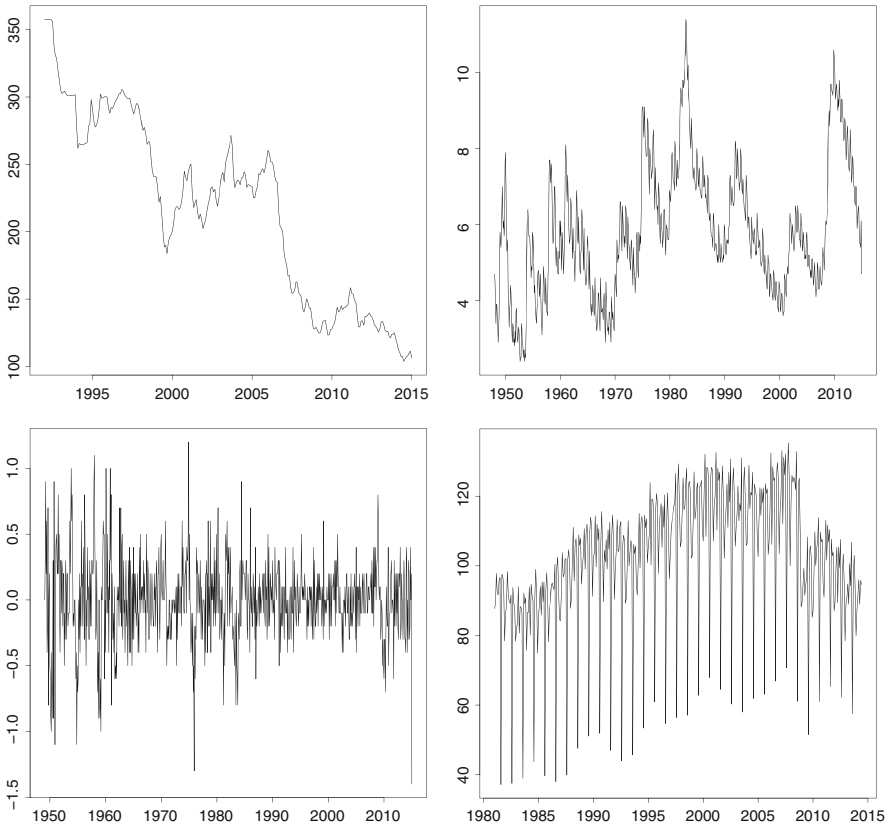

Fig. 1 Graphs of the time series employed in the empirical study

shows roughly an overall similar pattern, with spikes, irregular seasonality, and non-stationarity both in mean and variance. Such a similarity is roughly confirmed by the patterns of their *EACF*s (Fig. 2). Regarding the time series *TS*3–4, they exhibit more regular overall behaviors but deep differences in terms of their structures. In fact, by examining Figs. 1 and 2, it becomes apparent that unlike *TS*4, *TS*3 is roughly trend stationary with a 12-month seasonality with a persistence of the type moving average—according to the (unreported) Partial *EACF*—different from the one characterizing *TS*4, appearing to follow an autoregressive process. Regarding *TS*4, this time series has been included for being affected by two major problems: an irregular trend pattern with a significant structural break located in 2009 and seasonal variations with size approximately proportional with the local

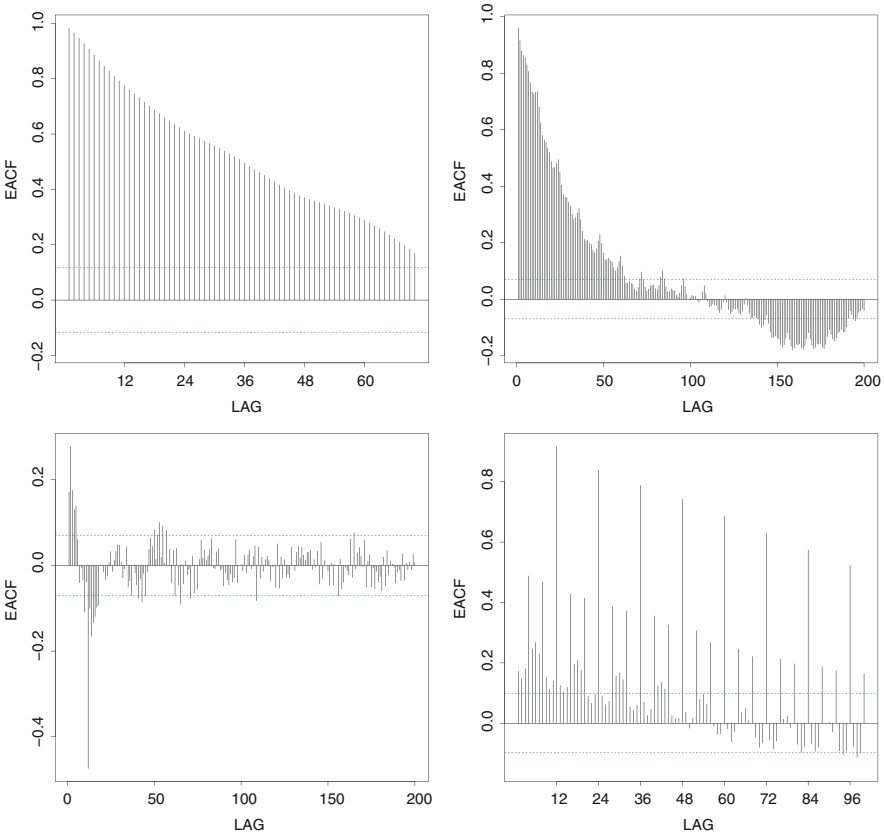

Fig. 2 *EACF*s for the time series employed in the empirical study

level of the mean. A potential source of nonlinearity, this form of seasonality is often dealt with by making it additive through ad hoc data transformation. However, this is not a risk-free procedure, for being usually associated with the critical task of back-transforming the data, as shown in [17, 18]. Quantitative assessment of the quality of the predictions generated by *MUNI* are made by means of the following three metrics—computed on the test sets of each of the four time series—i.e.: $\text{RMSE}^{(h)} = \sqrt{\frac{1}{s}\sum |x^S - \hat{x}^S|^2}$, $\text{MPE}^{(h)} = 100\frac{1}{s}\sum[\frac{x^S - \hat{x}^S}{x^S}]$, $\text{MAPE}^{(h)} = 100\frac{1}{s}\sum|\frac{x^S - \hat{x}^S}{x^S}|$, with s the length of x^S and h the number of steps ahead the predictions are evaluated, here $h = 1, 2, 3, 4$ (Figs. 3 and 4).

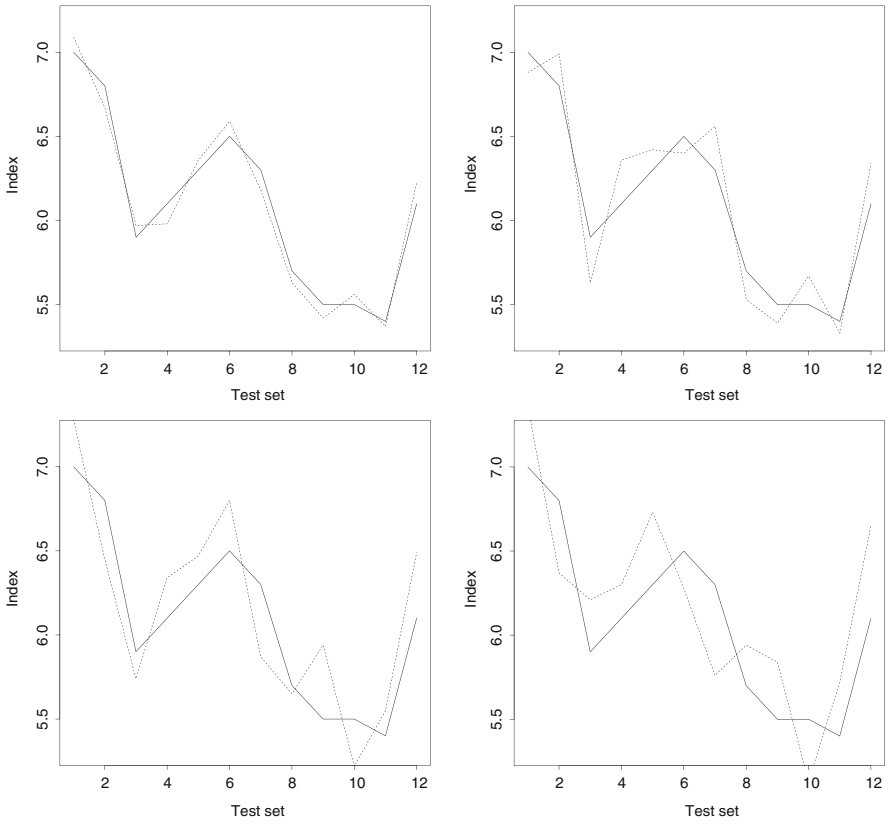

Fig. 3 *TS*1: test set. True (*continuous line*) and 1,2,3,4-step ahead predicted values (*dashed line*)

2.1 Results

As illustrated in Sect. 1.1.4, each of the employed *ANN*s has been implemented according to a variable-specific set $\boldsymbol{\Gamma}$, containing all the grids whose values are reported in Table 4. It is worth emphasizing that, in practical applications, not necessarily the set $\boldsymbol{\Gamma}$ encompasses the optimal (in the sense of the target function $\mathfrak{B}$) parameter values of a given network. More realistically, due to the computational constraints, one can only design a grid set able to steer the searching procedure towards good approximating solutions (Table 6). The outcomes of the empirical analysis outlined in the previous Sect. 2 are reported in Table 7. From its inspection, it is possible to see the good performances, with a few exceptions, achieved by the procedure. The series denominated *TS*4, in particular, shows a level of fitting that can

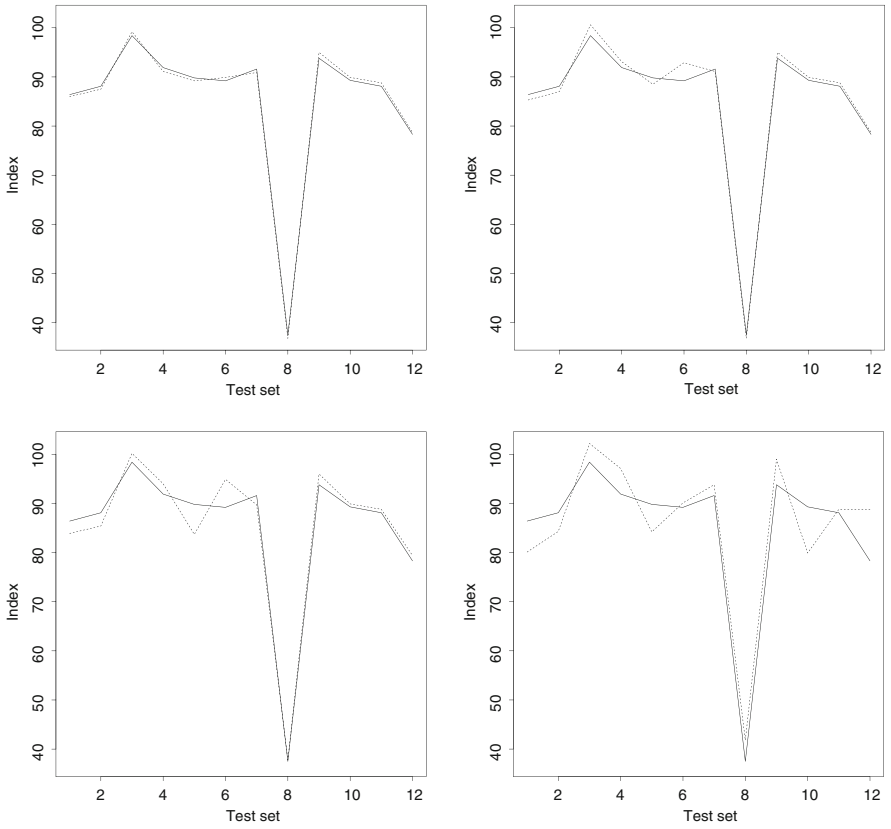

Fig. 4 *TS*4: test set. True (*continuous line*) and 1,2,3,4-step ahead predicted values (*dashed line*)

be considered particularly interesting, especially in the light of its moderate sample size and the irregularities exhibited by the lower frequency components, i.e., $d4$ and $s4$ (Fig. 5). The procedure chooses in this case relatively simple architectures: in fact (see Table 6), excluding $s4$ (with six lags and the parameter $\nu = 5$), for all the remaining components we have a more limited number of delays and of hidden neurons ($\nu \leq 3$). Regarding the performances, it seems remarkable the level of fitting obtained at horizon 1 and 2, for a *MAPE* respectively equal to 0.85 and 1.43, and a *RMSE* of 0.71 and 1.5. Visual inspection of Fig. 4 confirms this impression as well as the less degree of accuracy recorded at farther horizons, even though it appears how, especially horizon 3 predictions ($MAPE = 2.63$), can provide some insights about the future behavior of this variable. Other than from Table 7, less impressive performances can be noticed for $TS1$ by examining Fig. 3. However, it

Table 4 Grid values employed in the empirical analysis

Series	Parameters					
	β	ρ	α	η	ν	λ
*TS*1	3; 4; 5	300; 400	50; 100; 200	0.001; 0.01;0.1	1– 8	1–6; 12; 15; 18 21; 24; 30; 36
*TS*2	4; 5; 6	2; 6	50; 100; 200	0.0001; 0.001 ; 0.01; 0.1	1–10	1–12; 15; 18; 21; 24; 27 30; 33; 36; 48; 60; 72
*TS*3	4; 5; 6	2; 3	50; 100; 200	0.0001; 0.001; 0.01; 0.1	1–6	1–6; 8; 12; 18; 21 24; 36; 48
*TS*4	3; 4; 5	150; 200	50; 100; 200	0.001; 0.01; 0.1	1–6	1–12; 15; 18; 21; 24 27; 30; 33; 36; 48; 60

Table 5 Length of the subsets of the original time series

	Size			
Set	*TS*1	*TS*2	*TS*3	*TS*4
$\{x^A\}_t$	229	673	660	354
$\{x^V\}_t$	36	120	120	60
$\{x^S\}_t$	12	12	12	12

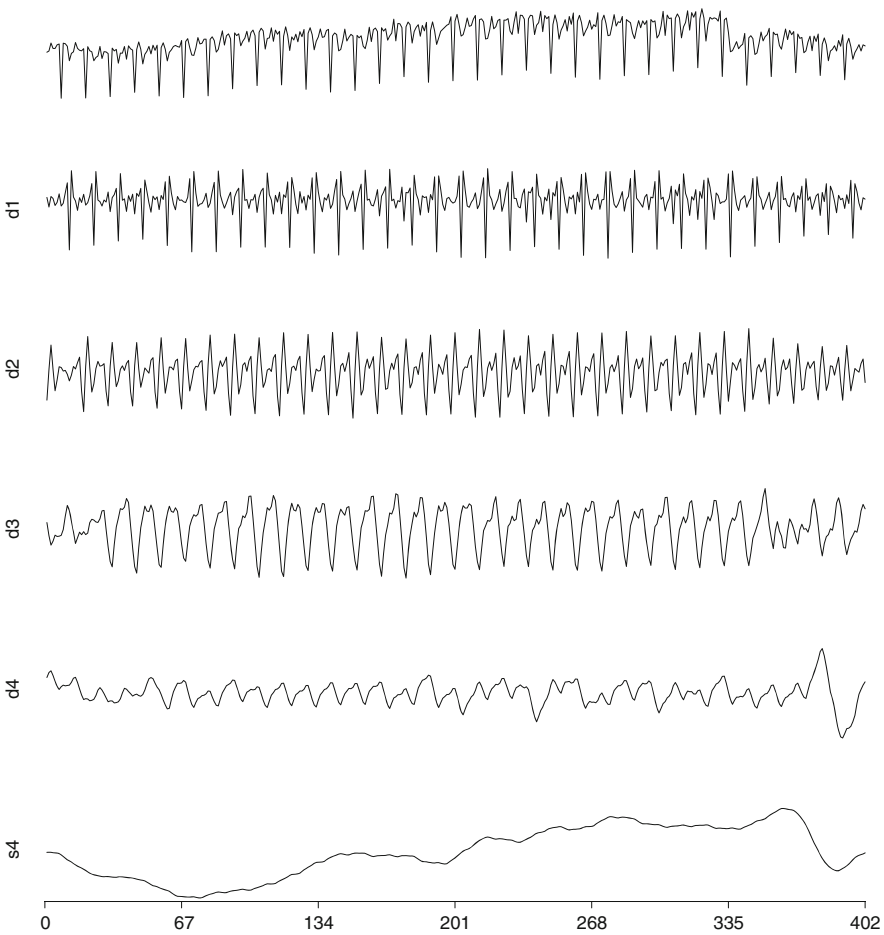

Fig. 5 *TS*4 and its *MODWT* coefficient sequence $d_{j,t}$; $j = 1, \ldots, 4$

Table 6 Parameters chosen by *MUNI* for each time series at each frequency component

Time series	Crystals	Parameters			
		α	η	λ	ν
TS1	d1	200	0.1	1-2-3-12-18-36	7
$\rho = 400$	d2	200	0.1	1-2-3-4-12-36	8
$\beta = 5$	d3	200	0.1	1-2-3-4-12-15-18	8
	d4	200	0.1	1-2-3-4-12-15-18-30	8
	s4	200	0.1	1-2-3-4-12-18-24-30	8
TS2	d1	100	0.001	1-2-12	4
$\rho = 7$	d2	100	0.001	1-2-12-18	5
$\beta = 6$	d3	100	0.001	1-2-3-30	4
	d4	50	0.001	1-2-4-48	3
	d5	100	0.001	1-2-12–30	4
	d6	100	0.01	1-4-36	2
	s6	100	0.01	1-2-3-10-60	4
TS3	d1	100	0.01	1-2-12	2
$\rho = 2$	d2	100	0.01	1-2-3-12	2
$\beta = 5$	d3	100	0.01	1-2-12-18	3
	d4	50	0.01	1-2-3-4-21	3
	s4	100	0.01	1-2-3-5-6-21	3
TS4	d1	100	0.001	1-2-3-12	2
$\rho = 150$	d2	100	0.001	1-2-12	2
$\beta = 5$	d3	100	0.001	1-2-5-15	3
	d4	200	0.01	1-3-4-24	3
	s4	200	0.1	1-3-15-18-36-48	5

Table 7 Goodness of fit statistics computed on the test set for the four time series considered

	Horizon	RMSE	MPE	MAPE		Horizon	RMSE	MPE	MAPE
TS1	1	1.729	0.303	1.485	*TS2*	1	0.093	−0.105	1.424
	2	2.705	−0.097	2.354		2	0.186	0.408	2.829
	3	4.71	1.063	3.56		3	0.294	0.75	4.382
	4	7.184	1.588	6.133		4	0.375	1.361	5.835
TS3	1	0.129	−0.677	1.841	*TS4*	1	0.712	−0.047	0.849
	2	0.215	0.712	3.224		2	1.507	0.351	1.43
	3	0.224	−0.694	3.12		3	2.908	0.035	2.626
	4	0.295	2.693	4.075		3	5.591	0.807	5.938

Values for *TS3* obtained by back-transformation

has to be said that, among those included in the empirical experiment, this time series proves to be the most problematic one both in terms of sample size and for exhibiting a multiple regime pattern made more complicated by the presence of heteroscedasticity. Such a framework induces *MUNI* at selecting architectures which are too complex for the available sample size. As reported in Table 6, in fact, the number of hidden neurons is large and reaches, for almost all the frequency components chosen ($\beta = 5$), its maximum grid value ($\nu = 8$). Also, the number of input lags selected is always high, whereas the regularization parameter reaches, for all the decomposition levels, its maximum value ($\eta = 0.1$). Such a situation is probably an indication of how the procedure tries to limit model complexity by using the greatest value admitted for the regularization term, nevertheless the selected networks still seem to over-fit. This impression is also supported by the high number of iterations (the selected value for α is 200 for each of the final networks) which might have induced the selected networks to learn irrelevant patterns. As a result, *MUNI* is not able to properly screen out undesired, noisy components, which therefore affect the quality of the predictions. However, notwithstanding this framework, the performances can be still regarded as acceptable considering the predictions at lag 1 and perhaps at lag 2 (*RMSE* = 1.73 and 2.70, respectively), whereas they significantly deteriorate at horizon 3, where the *RMSE* reaches the value of 4.71. Horizon 4 is where *MUNI* breaks down, probably for the increasing degree of uncertainty present at higher horizons associated with poor network generalization capabilities. With an *RMSE* of 7.18, that is, more than 4 times higher than horizon 1 and an *MPE* of 1.59 (>5 times), additional actions would be in order, e.g., increasing the information set by including ad hoc regressors in the model. As already mentioned, *TS*2 shows an overall behavior fairly similar to *TS*1, in terms of probabilistic structure, non-stationarity components and multiple regime pattern. However, the more satisfactory performances recorded in this case are most likely to be connected to the much bigger available sample size. In particular, it is worth emphasizing the good values of the *MAPE* for the short term predictions ($h = 1, 2$), respectively, equal to 1.42 and 2.83 as well as the *RMSE* obtained at horizon 4, which amounts to 0.37. In this case, more parsimonious architecture are chosen (see Table 6) for the $\beta = 6$ selected number of components the original time series has been broken into, with a number of neurons ranging from 2 (for the crystal $d6$) to 5 (for the crystals $d2$), which are associated with input sets of moderate size, ($\lambda = 5$ is the max value selected). As a pre-processed version of *TS*2, *TS*3 shows a more regular behavior (even though a certain amount of heteroscedasticity is still present) with weaker correlation structures at the first lags and a single peak at the seasonal lag 12 (*EACF* = −0.474). As expected, *MUNI* for this case selects simpler architectures for each of the $\beta = 5$ sub-series, with a limited number of input lags and a smaller number of neurons ($\nu \leq 3$). Although generated by more parsimonious networks, the overall performances seem to be comparable to those obtained in the case of *TS*2. In fact, while they are slightly worse for the first two lags (*MAPE* = $1.84, 3.22$ for $h = 1, 2$ respectively versus 1.42 and 2.83), the error committed seems to decrease more smoothly as the forecast horizon increases. In particular at horizon 4, *MUNI* delivers better predictions than in the case of

the un-transformed series: in fact, the recorded values of *RMSE* and *MAPE* are respectively of 0.29 and 4.07 for *TS*3 and 0.37 and 5.83 for *TS*2.

Acknowledgements The author is deeply indebted to Professor Dimitris N. Politis of the Applied Physics and Mathematics Department of the University of California San Diego, for his valuable research assistance. Computational support provided by Mr. Wilson Cheung of the same Department is also gratefully acknowledged.

References

1. De Gooijer, J.G., Hyndman, R.J.: 25 years of time series forecasting. Int. J. Forecast. **22**, 443–473 (2006)
2. Mallat, S.G.: A theory for multiresolution signal decomposition: the wavelet representation. IEEE Trans. Pattern Anal. Mach. Intell. **11**, 674–693 (1989)
3. Mallat, S.: A Wavelet Tour of Signal Processing. Academic, New York (1999)
4. Akansu, A.N., Haddad, R.A.: Multiresolution Signal Decomposition: Transforms, Subbands, and Wavelets. Academic, New York (2001)
5. Patra, J., Pal, R., Chatterji, B., Panda, G.: Identification of nonlinear dynamic systems using functional link artificial neural networks. IEEE Trans. Syst. Man Cybern. B Cybern. **29**, 254–262 (1999)
6. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, New York, NY (1995)
7. Farge, M.: Wavelet transforms and their applications to turbulence. Annu. Rev. Fluid Mech. **24**, 395–458 (1992)
8. Daubechies, I., et al.: Ten lectures on wavelets. In: Society for Industrial and Applied Mathematics, vol. 61. SIAM, Philadelphia (1992)
9. Qian, T., Vai, M.I., Xu, Y.: Wavelet Analysis and Applications. Springer, Berlin (2007)
10. Percival, D.B., Walden, A.T.: Wavelet Methods for Time Series Analysis, vol. 4. Cambridge University Press, Cambridge (2006)
11. Aboufadel, E., Schlicker, S.: Discovering Wavelets. Wiley, New York (2011)
12. Härdle, W., Kerkyacharian, G., Picard, D., Tsybakov, A.: Wavelets, Approximation, and Statistical Applications, vol. 129. Springer, Berlin (2012)
13. Webb, A.R.: Statistical Pattern Recognition. Wiley, New York (2003)
14. Pal, S.K., Mitra, S.: Multilayer perceptron, fuzzy sets, and classification. IEEE Trans. Neural Netw. **3**, 683–697 (1992)
15. Michie, D., Spiegelhalter, D.J., Taylor, C.C.: Machine Learning, Neural and Statistical Classification. Horwood Ellis, Ltd., Hempstead (1994)
16. Box, G.E.P., Jenkins, G.M.: Time Series Analysis, Forecasting, and Control, vol.136. Holden-Day, San Francisco (1976)
17. Chatfield, C., Prothero, D.: Box-Jenkins seasonal forecasting: problems in a case-study. J. R. Stat. Soc. Ser. A (General) (1973) 295–336
18. Poirier, D.J.: Experience with using the Box-Cox transformation when forecasting economic time series: a comment. J. Econ. **14**, 277–280 (1980)