

Improving DWT-RNN model via B-spline wavelet multiresolution to forecast a high-frequency time series

Zeinab Hajiabotorabi^{a,b}, Aliyeh Kazemi^c, Faramarz Famil Samavati^d,
Farid Mohammad Maalek Ghaini^{a,*}

^a Department of Mathematics, Faculty of Science, Yazd University, Yazd, Iran

^b Department of Mathematics, Faculty of Science, Payame Noor University, Yazd, Iran

^c Department of Industrial Management, Faculty of Management, University of Tehran, Tehran, Iran

^d Department of Computer Science, University of Calgary, Calgary, Alberta, Canada



ARTICLE INFO

Article history:

Received 14 April 2019

Revised 24 July 2019

Accepted 25 July 2019

Available online 25 July 2019

Keywords:

Discrete wavelet transform

B-spline wavelets multiresolution

Artificial neural networks

Return volatility

Financial time series forecasting

ABSTRACT

This paper presents a recurrent neural network (RNN) which is improved by using an efficient discrete wavelet transform (DWT) for predicting a high-frequency time series. In the combined DWT-RNN model, first, a multiresolution based on B-spline wavelet of high order d (BSd) is used to decompose the time series into several smooth data sets. Therefore, an approximation data set (with low-frequency) and several detail data sets (with high-frequency), with small wave amplitude, are obtained. Then, all decomposed components are used as RNN inputs. The proposed BSd-RNN model can approximate smooth patterns with satisfactory accuracy, and because of the local properties, BSd is a better choice than other common DWT such as Haar and Daubechies of order n (dbn), for preprocessing the high-frequency time series. According to results of performance metrics for predicting four different stock indices, the BSd-RNN model outperforms other common DWT-RNN model such as Haar-RNN and dbn-RNN. Also, the results show the BSd-RNN model outperforms other common artificial neural network (ANN) model such as multilayer feed-forward neural network (FFNN). Finally, The results show that BS3-RNN predicting model has better predictive ability than other compared models which use other wavelets or other ANNs.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Nowadays, there are many models to predict various types of data specially predicting of time series. Since the most prediction models of time series are based on the behavior of historical data, analysts focus on efficient data-driven models for real-time forecasting. Artificial neural networks (ANNs) as non-parametric model are often successfully used for predicting time series (Kazemi, Shakouri, Menhaj, Mehregan, & Neshat, 2010; Shaghaghi, Bonakdari, Gholami, Ebtehaj, & Zeinolabedini, 2017; Vaisla, 2010; Wang, Zou, Su, Li, & Chaudhry, 2013). They are ideal especially when we do not have any other description of the observed series. An ANN is trained from the historical data with the hope that it will discover hidden dependencies and be able to predict the future. ANNs are better suited to recognize nonlinear relationships of high-frequency time series (Bento, Pombo, Calado, &

Mariano, 2018). Recurrent neural networks (RNNs) have feedback connections inside the neural network to learn temporal patterns. Therefore, RNNs can be used for effective sequential data modeling and time series analyses (Taha & Taha, 2018). Moreover, wavelet transform (WT) is another commonly used feature to better deal with high fluctuation and variance of time series (Huang & Wang, 2018; Nguyen & He, 2015; Olsen & Samavati, 2008; Wang, Wang, Zhang & Guo, 2011). WT of time series presents a better behavior typically with low volatility (more stable variance) than the original time series, thus forecasting is in a better performance (Conejo, Plazas, Espinola, & Molina, 2005).

For the first time, Zhang and Benveniste (1992), and then Krishnaprasad and Pati (1993) proposed that ANN performance can be improved by using WT to generate lower frequency samples as inputs of the ANN. It is noteworthy to mention that both continuous wavelet transforms (CWT, e.g., Fourier transforms (Mallat, 1999)), and discrete wavelet transforms (DWT, e.g., Haar transforms (Haar, 1985) and Daubechies of order n (dbn) transforms (Daubechies, 1992)) work partly well for preprocessing of ANN inputs. Researchers have commonly hybridized WT, and ANN as WT-ANN forecasting models to improve the higher accuracy of

* Corresponding author.

E-mail addresses: z_habotorabi@pnu.ac.ir (Z. Hajiabotorabi), aliyehkazemi@ut.ac.ir (A. Kazemi), samavati@cpsc.ucalgary.ca (F.F. Samavati), maalek@yazd.ac.ir (F.M. Maalek Ghaini).

performance (Altunkaynak & Ozger, 2016; Bento et al., 2018; Chandar, Sumathi, & Sivanandam, 2016; Doucoure, Agbossou, & Cardenas, 2016; Homayouni & Amiri, 2011; Hsieh, Hsiao, & Yeh, 2011; Khuat, Le, Nguyen, & Le, 2016; Krishnaprasad & Pati, 1993; Lahmiri, 2014; Li, Li, & Wang, 2016; Mishra, Soni, & Sharma, 2018; Qu, Mao, Zhang, Zhang, & Li, 2019; Shi, Wang, Jiang, & Liu, 2018; Zhang & Benveniste, 1992).

Fourier (1807) with his theories in the field of frequency analysis and using scale varying basis functions and computing the energy of a function in 1930s, Grossmann and Morlet (1984), Mallat (1999), Meyer (1992), and Daubechies (1992) with their works on wavelet applications, all have importance roles to develop WT.

B-spline wavelet of order d (BSd) is defined as a local DWT. When $d = 1$ (BS1), it is known as Haar WT with discontinuous wavelets, while $d > 2$, it is generated by smooth wavelets. The B-spline wavelets are often chosen as good scaling functions since their matrix filters are very simple and efficient (Samavati & Bartels, 1999; Bartels & Samavati, 2000, 2011). Conventionally, B-spline wavelets of higher orders are constructed with the goal of semi-orthogonality, which results in the analysis of full matrices (Samavati & Bartels, 1999). An alternative approach for generating multiresolution matrices is the reverse subdivision, which has been originally introduced by Samavati and Bartels (1999), Bartels and Samavati (2000). By using this approach, it is possible to obtain banded matrices for biorthogonal B-spline wavelets, whose bands are narrower than the ones conventionally produced by B-spline wavelets. Olsen & Samavati used the reverse subdivision as a discrete approach for constructing a scheme of different curves and surfaces (Olsen & Samavati, 2008; Samavati, Amiri, & Bartels, 2002).

Multiresolution analysis has proven itself as a very compelling mathematical framework in many applications (Stollnitz, Deroose, & Salesin, 1996; Unser, 1997). Decomposition of a high-resolution signal to low resolution and details signals, for multiple times, provides an effective tool for better compression and also model synthesize (Finkelstein & Salesin, 1994; Samavati, Bartels, & Olsen, 2007; Stollnitz et al., 1996). Brunn, Sousa, and Samavati (2007) exploited multiresolution to achieve curve synthesis by capturing and reusing artistic styles. Wecker, Samavati, and Gavriloa (2010) used multiresolution in iris synthesis and contextual void patching in terrains. Alderson and Samavati (2014) applied reverse subdivision for optimizing line-of-sight queries in real-time navigation in large terrains. Alderson, Mahdavi-Amiri, and Samavati (2018) used multiresolution for the offsetting spherical curves in vector and raster form. Moltaji, Runions, and Samavati (2017) utilized subdivision and multiresolution for partition of unity parametrics (PUPs).

In contrast to Haar as BS1 (Chandar et al., 2016), BSd of higher order $d > 1$ has been rarely used for preprocessing of a time series in the predicting ANN model. In this paper, we employ BSd in order to achieve a more accurate predicting ANN models such as RNN. We particularly employ reverse subdivision multiresolution filters (Bartels & Samavati, 2000; Samavati & Bartels, 1999) in Matlab code to be implemented to let ANNs act stronger in prediction. The new combined model is named BSd-RNN, in which BSd is used due to its local and smoothness properties. In this way, first by applying a BSd multiresolution, the original time series is decomposed into different frequency levels of resolution data sets that all of them are smooth. By multiresolution of higher order on the high-frequency time series, a very fitted smooth approximate data set, and the several smooth detail data sets which are of high frequencies with small wave amplitudes are obtained. Then, we use all smooth decomposed data set as inputs of the network in BSd-RNN model to predict a high-frequency time series. As far as we searched, it is the first time that the efficient BSd multiresolution of higher order, based on reverse subdivision, is used for en-

abling the ANN model to forecast the high-frequency time series, and with more accuracy than similar common models.

The rest of the paper is organized as follows: Section 2 describes background and methods in detail. In this section, we introduce DWT, BSd multiresolution based on reversing subdivision, two types of ANNs (RNN and feed-forward neural network (FFNN)), autoregressive integrated moving average (ARIMA), and generalized autoregressive conditional heteroskedasticity (GARCH) models. Performance metric is also introduced in this section. The application and evaluation of the proposed BSd-RNN model for predicting a volatility time series are presented in Section 3. In this section, the capability of the proposed BSd-RNN model in predicting the volatility of some different index stock markets is evaluated by performance metric. Section 4 summarizes the main results and discusses the reliability of the proposed BSd-RNN model compared to another type of neural network, BSd-FFNN and other common forecasting DWT-RNN models which use other DWT wavelets such as Haar and Daubechies. Finally, Section 5 describes conclusions.

2. Background and methods

The proposed prediction process consists of two steps: (1) The original time series $s(t)$ is preprocessed with some BSd, $d = 1, 2, 3, 4$. (2) All decomposed data sets are fed as the inputs of the RNN model. For comparison purposes, some common DWT-ANN are used as predictors consist of DWT-FFNN, Haar-RNN and dbn-RNN ($n = 3, 4$). In addition, the non-parametric proposed BSd-RNN model is compared with ARIMA and GARCH models. Therefore, in the following, DWT, ANN, ARIMA, and GARCH models are described in more detail.

2.1. Discrete wavelet transform

DWT is a discrete implementation of the CWT, and just as accurate as CWT, but more efficient (Altunkaynak & Ozger, 2016). The idea behind DWT is based on the multiresolution analysis (MRA), where the original time domain is decomposed into several other scales with different levels of resolution, in a process known as multiresolution decomposition. Two related functions ϕ_{ik} and ψ_{ik} represent the scaling and wavelet functions as follows:

$$\phi_{ik}(t) = 2^{-i/2} \phi(2^i t - k) \quad (1)$$

$$\psi_{ik}(t) = 2^{-i/2} \psi(2^i t - k) \quad (2)$$

where i and k are the scaling and translation variables, respectively. Fig. 1 shows the MRA processing on time series $s(t)$, $t \in \{t_1, t_2, \dots, t_T\}$ which is decomposed into an approximation coefficients vector $a(t)$ by using a low-pass (LP) filter and a detail coefficients vector $d(t)$ by using high-pass (HP) filter in the wavelet domain. The process of decomposition is executed iteratively on the approximation coefficients of the previous level, to calculate both detail and approximation coefficients of the next level. Therefore after j levels of decomposition, $s(t)$ is decomposed into several parts, $a_1(t), a_2(t), \dots, a_j(t)$, approximation data sets with lower resolutions, and $d_1(t), d_2(t), \dots, d_j(t)$, detail data sets. The entries $a_i(t) = (a_{ik})_k = \langle s(t) \cdot \phi_{ik}(t) \rangle$ are the coefficients which generate the following sequence of approximation curves in the nested functional subspaces $V_j \subset V_{j-1} \dots \subset V_1 \subset V_0$, where $\langle \cdot \rangle$ is the dot product of two vectors.

$$g_i(t) = \sum_k a_{ik} \phi_{ik}(t), \quad i = 1, 2, \dots, j \quad (3)$$

and the detail coefficient $d_i(t) = (d_{ik})_k = \langle s(t) \cdot \psi_{ik}(t) \rangle$ generates the following sequence detail signal:

$$h_i(t) = \sum_k d_{ik} \psi_{ik}(t), \quad i = 1, 2, \dots, j \quad (4)$$

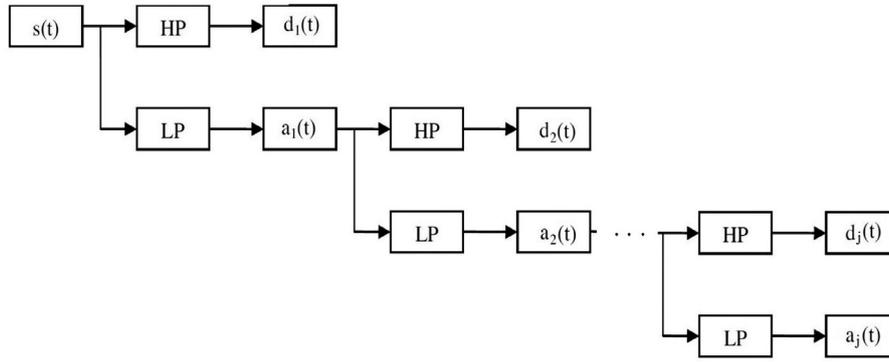


Fig. 1. The MLR processing in j levels on time series $s(t)$.

Thus, the original $s(t)$ time series can be reconstructed as:

$$s(t) \approx g_1(t) + \sum_{i=1}^j h_i(t) \tag{5}$$

By decomposing a high-frequency time series $s(t)$ into a set of better-behaved time series $g_1(t)$ and constitutive time series $h_i(t), i = 1, 2, \dots, j$ with less fluctuation, time series forecasting will have higher accuracy (Mandal, Haque, Meng, Srivastava, & Martinez, 2013). Therefore, in the proposed DWT-RNN model, all decomposed components of MRA on original time series, $g_1(t)$ and $h_i(t), i = 1, 2, \dots, j$ are simultaneously used as the input of the RNN. There is no theory concerning the determination of the best level of decomposition. It is suitable using of the $j = \lceil \log(T) \rceil$ levels of decomposition, where T is the current time in the time series (Altunkaynak & Ozger, 2016).

Each Haar wavelet $\psi_{ik}(t)$ has a zero average over its support $[2^i n, 2^i(n+1)]$. If f is locally regular and 2^i is small, then it is nearly constant over this interval and the wavelet coefficient $\langle f, \psi_{ik} \rangle$ is thus nearly zero. This means that large wavelet coefficients are located at sharp signal transitions only. With a jump in time, the story continues, when Stromberg (1981) found a piecewise linear function ψ that also generated an orthonormal basis and gave better approximations of smooth functions. The systematic theory for constructing orthonormal wavelet bases was established by Meyer (1992), and Mallat (1999). It was inspired by original ideas developed in computer vision by Adelson, Simoncelli, and Hingorani (1987) to analyze images at several resolutions. Digging more into the properties of orthogonal wavelets and multiresolution approximations brought to light a surprising relation with filter banks constructed with conjugate mirror filters, and a fast wavelet transform algorithm decomposing signals with $O(N)$ operations (Mallat, 1999).

2.2. Bsd multiresolution based on reversing subdivision

In this article, we adopt a notation for representing Bsd multiresolution operation in simple matrix forms which has been found by Samavati and Bartels (1999), and Samavati et al. (2002). Bsd is used in decomposition and reconstruction of any curves and surfaces in computer graphic models. They have denoted these matrices by A_i and B_i as bounded LP filters, and P_i and Q_i as bounded HP filters, where the subscript $i = 1, 2, \dots, j$ is used as the current level of resolution, and the relationship of the filters is as follows:

$$\begin{bmatrix} A_i \\ B_i \end{bmatrix} = [P_i | Q_i]^{-1} \tag{6}$$

In the matrix form of Bsd, the vector a_{i-1} is decomposed into two parts: a_i (approximation coefficients) and d_i (detail coefficients)

which are generated by using filters A_i and B_i for $i = 1, 2, \dots, j$ as follows:

$$a_i = A_i \cdot a_{i-1} \tag{7}$$

$$d_i = B_i \cdot a_{i-1} \tag{8}$$

In addition, the following formulas are used for the reconstruction of a_{i-1} :

$$a_{i-1} = P_i \cdot a_i + Q_i \cdot d_i \tag{9}$$

According to (3), (4), and (9), for Bsd multiresolution of time series $s(t), t \in [t_1, t_T], a_0 = (s(t_1), s(t_2), \dots, s(t_T))$ represent by reversing subdivision in j decomposition levels as:

$$s(t) = a_0 = g_1(t) + \sum_{i=1}^j h_i(t) \tag{10}$$

Therefore, $s(t)$ is obtained by reconstructed relationship (10).

2.3. Artificial neural network

ANN is a computational method which is designed to imitate a collection of neuron connections in the brain. Recently, ANNs have been developed their application in complex problems in the world. There are many types of ANN architectures; however, here, only two types of networks are called FFNN and RNN are discussed.

2.3.1. Feed-forward neural network

FFNN consists of input, hidden, and output layers (see Fig. 2). (i) The input layer takes normalized input vectors of data for sending to the hidden layers, (ii) one or more consecutive hidden layers containing neurons to capture the nonlinearity in the samples, (iii) an output layer consists of one or more neurons which representing the dependent variables. In Fig. 2, the FFNN structure in which an e_1 dimensional input vector x_n gives an output value y_n , where $n = 1, 2, \dots, N$, is shown. In each layer, every neuron response is given by the non-linear activation function $f(\cdot)$ which enables the system to learn nonlinear relationships, with a cost given by a biased weighted sum (Bento et al., 2018). These weights are coefficients (parameters) of the ANN model. The size of each weight represents the relative strength of the connection. Considering an FFNN with M hidden layers, the relationship between the output y_n and the input x_n can be expressed by the following relation:

$$y_n = f(W^{(M+1)} \cdot f(W^{(M)} \cdot \dots \cdot f(W^{(2)} \cdot f(W^{(1)} \cdot x_n + b_1) + b_2) + \dots + b_M) + b_{M+1}) \tag{11}$$

where $W^{(i)} \in \mathbb{R}^{e_i}$, $i = 1, 2, \dots, M + 1$, e_i is the number of neurons in layer i , and b_i is the bias.

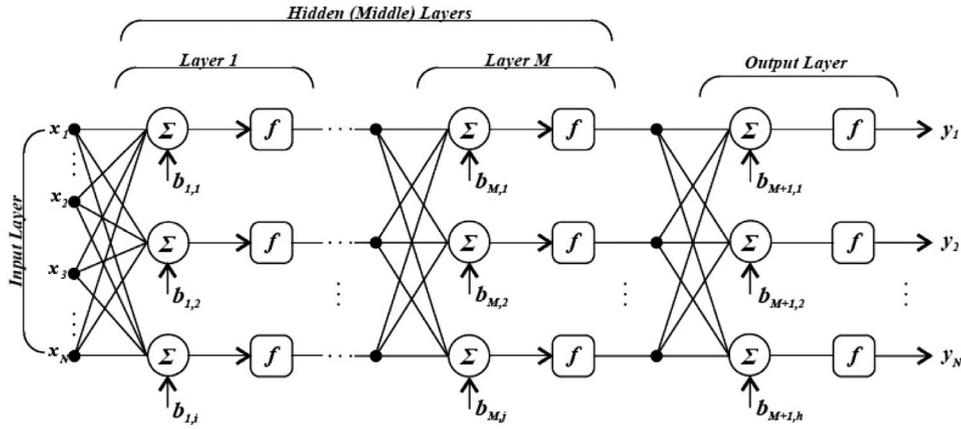


Fig. 2. The FFNN structure.

To build a model for forecasting a time series $s_t = s(t) \in [t_1, t_T]$ at l steps ahead, the network is processed through three sections (Kazemi et al., 2010): (1) The training section, where the network is trained to predict future data based on the past and present data. (2) The test section, where the network is tested to stop training or to continue training. (3) The evaluation section, where the network ceases training and is used to forecast future data and to calculate different measures of error. These networks are so smart, and they can be successfully used for forecasting. This model uses supervised learning which is a machine learning task for learning a function that maps an input x_n as follows:

$$x_n = (s_n, s_{n+1}, \dots, s_{n+e_1-1}), n = 1, 2, \dots, N \quad (12)$$

where, the sample number is $N = T - e_1 + 1 - l$ which are divided into the training, test and evaluating samples, and a given target output z_n as follows:

$$z_n = s_{n+e_1-1+l} \quad (13)$$

The most important aspect of an ANN model is its learning ability. The learning problem turns out to be finding the optimum weights from a given input samples x_n by updating the weight vector in order to minimize the error, i.e., the difference between the target output response z_n and the network output y_n (through a performance function). Commonly, the chosen performance metric is the mean squared error (MSE), given by the following expression:

$$MSE = \frac{1}{N} \sum_{n=1}^N (y_n - z_n)^2 \quad (14)$$

A good choice for training a network is the conjugate gradient (CG) algorithm. CG avoids a time-consuming line-search per learning epoch, requires less memory usage, which makes the algorithm faster than any second order algorithms. CG can train any network which has differentiable activation functions. A full description of CG algorithm implementation can be found in Moller (1993).

2.3.2. Recurrent neural network

Unlike FFNNs, RNNs can use their internal state to process sequences of inputs. In a RNN with hidden layer m_n (see Fig. 3), the relationship between the output y_n and the input x_n can be expressed by the following relations (Elman, 1990):

$$m_n = f(Wx_n + Um_{n-1} + b_m) \quad (15)$$

$$y_n = f(Vm_n + b_y) \quad (16)$$

where $f(\cdot)$ is an activation function such as $\tanh(\cdot)$ (Xing, Cambria, & Zhang, 2019). Also, W , U , V and b are parameters matrices and vector.

2.4. Autoregressive integrated moving average models

The general ARIMA(p, r, q) model is given by Box, Jenkins, Reinsel, and Ljung (2015). This model is one of the most general class of models for forecasting a time series which can be made to be stationary by differencing. More precisely, ARIMA model is generalized from autoregressive moving average (ARMA) model in which the assumption on stationary of time series is not necessary. The important characterization of ARIMA model is that the predictions of the behaviour of a time series in the future depend on the past observations by a linear function and random errors, i.e., the ARIMA equation for forecasting a stationary series Y_t has the following form:

$$\phi(B) \nabla^r Y_t = \theta(B) \epsilon_t \quad (17)$$

where $r \geq 1$ is the degree of differencing, $\nabla = 1 - B$ is the differencing operator, the lag operator B , is defined as $BY_t = Y_{t-1}$, the operator which gives the previous value of the series. $\phi(B)$ and $\theta(B)$ are polynomials of degree p and q in B as below:

$$\phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p \quad (18)$$

$$\theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q \quad (19)$$

2.5. Generalized autoregressive conditional heteroskedasticity model

GARCH model is a very popular model which is used alone or in combination with other models in financial literature. Donaldson and Kamstra (1997) were one of the first demonstrated that neural networks could capture the nonlinear effects of volatility that GARCH models and their derivatives are not capable of modeling. Since this work, there have been several studies of hybrid models such as Bahrammirzaee (2010), Bildirici and Ersin (2013), Kristjanpoller and Minutolo (2016), Kristjanpoller and Hernandez (2017), Luo, Zhang, Xu, and Wang (2018), Kim and Won (2018), and Xing et al. (2019). This study differs from the previous ones as it is the first high order B-spline wavelet hybrid application to predict the return volatility of a stock index market.

Given the characteristic of heteroskedasticity of economic and financial time series, the autoregressive conditional heteroskedasticity (ARCH) models and their generalized (GARCH) are established for modeling by Engle (1982), and Bollerslev (1986). The GARCH(p, q) considers that the current conditional variance $\sigma_t^2 = \text{Var}(r_t | \mathcal{F}_{t-1}) = \mathbb{E}[(r_t - \mu_t)^2 | \mathcal{F}_{t-1}]$ (where \mathcal{F}_t is the information set (σ -field) of all information through the time t and $\mu_t = \mathbb{E}(r_t | \mathcal{F}_{t-1})$ is the conditional average of return) depends on p past conditional variances and on q past squared innovations (error terms). The

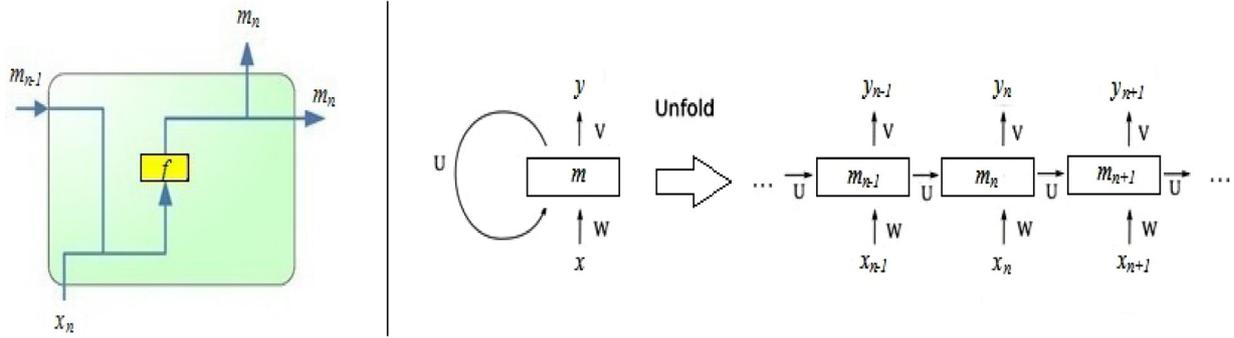


Fig. 3. The RNN structure.

continuously compounded rate of asset return from time $t - 1$ to t as follows:

$$r_t = 100 \times (\ln(p_t) - \ln(p_{t-1})) \tag{20}$$

Relation (20) is calculated as the first difference of the $\ln(p_t)$, where p_t is the asset price at time t . Then, the GARCH(p, q) model can be written as:

$$r_t = \mu_t + a_t \tag{21}$$

$$a_t = \sigma_t \eta_t \tag{22}$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i \sigma_{t-i}^2 + \sum_{j=1}^q \beta_j a_{t-j}^2 \tag{23}$$

where η_t is a sequence of independent and identically distributed random variables with zero mean and unit variance, σ_t^2 is the conditional variance of η_t and α_i, β_j , and α_0 are unknown coefficients to be estimated.

2.6. Performance metric

Several error metrics such as forecasting root mean square error (FRMSE), forecasting mean absolute error (FMAE), and forecasting mean absolute percentage error (FMAPE) are used to measure the performance of the models in predicting the time series datasets. These performance metrics are defined as (Willmott & Matsuura, 2005):

$$FRMSE = \sqrt{\frac{1}{N} \sum_{k=1}^N (\sigma_k^2 - \hat{\sigma}_k^2)^2} \tag{24}$$

$$FMAE = \frac{1}{N} \sum_{k=1}^N |\sigma_k^2 - \hat{\sigma}_k^2| \tag{25}$$

$$FMAPE = \frac{1}{N} \sum_{k=1}^N \left| \frac{\sigma_k^2 - \hat{\sigma}_k^2}{\sigma_k^2} \right| \times 100 \tag{26}$$

where σ_k is the actual volatility of the asset, $\hat{\sigma}_k$ is the respective predicted volatility, and N is the number of data.

3. Application and evaluation of the proposed BSd-RNN model

Prices of different financial assets such as currencies and stocks are constantly fluctuating as traders buy and sell these assets. The variation in the return prices over a period of time is called volatility. The volatility tells us about how turbulent the return price is and is an indicator of the risk involved. A stock with high volatility

involves high risk, but is also seen as an opportunity to make profits by the traders. Therefore, forecasting the volatility is important to trade in financial markets.

Time series analysis is one of the most challenging topics for financial researchers. In particular, the prediction of asset price, price return and return volatility time series, which needs a special analysis of trend volatility by focusing on historical data of the market, is a difficult problem. On the one hand, most of the forecasting models assume that there is some underlying stability in the market, and it is such that the future will be like the past. So, an ANN model is a good choice (Khuat et al., 2016; Vaisla, 2010; Wang et al., 2013). Moreover, due to the existence of non-stationary, high fluctuations and chaotic properties in the financial time series, they are known as the high-frequency time series (Vaisla, 2010; Wang et al., 2011). Fig. 4 shows the charts of some high-frequency time series such as the price return and the return volatility of the S&P 500 index market from 2000 to 2019, which is available in yahoo finance website (<https://finance.yahoo.com>). We Apply BSd to improve the RNN model for predicting some high-frequency financial time series such as the return volatility of several stock indices.

3.1. Data

In this section, we investigate the daily closing prices of the indices of the S&P 500, NASDAQ, Dow Jones Industrial Average (DJIA), and NYSE. All the historical prices have been downloaded from yahoo finance website, which is a standard reference for such purposes. We have considered the prices for the period from Jun 11, 2000, to Jun 11, 2019. In this period, there were about 5000 closing prices. The historical index price was used to calculate the monthly volatility time series.

The historical volatility (HV; $\sigma_t^2, t \in [t_1, t_T]$) of every index represented as the variance of the price return in (20). In particular, the HV is analyzed at 22 days, which is the volatility of a month with daily data. Then, we selected 80% of HV samples ($N = T - e_1 + 1 - l$) as the training, 15% as test, and the rest as evaluating samples. For the learning method, we chose the CG algorithm, and the learning rate and the momentum parameter were arbitrarily set at 0.1 and 0.9, respectively. The training of the ANN would stop when the MSE error becomes less or equal to 10^{-5} , or when the number of epochs reaches 10^5 .

3.2. BSd multiresolution on data set

Mentioned above, a predicting RNN model is employed to predict the HV time series $\sigma^2(t)$, which show all the HVs in time interval $[t_1, t_T]$. We would like to predict $\sigma^2(t_{T+l})$, at l steps ahead. For this purpose, BSd is applied for multiresolution of the financial

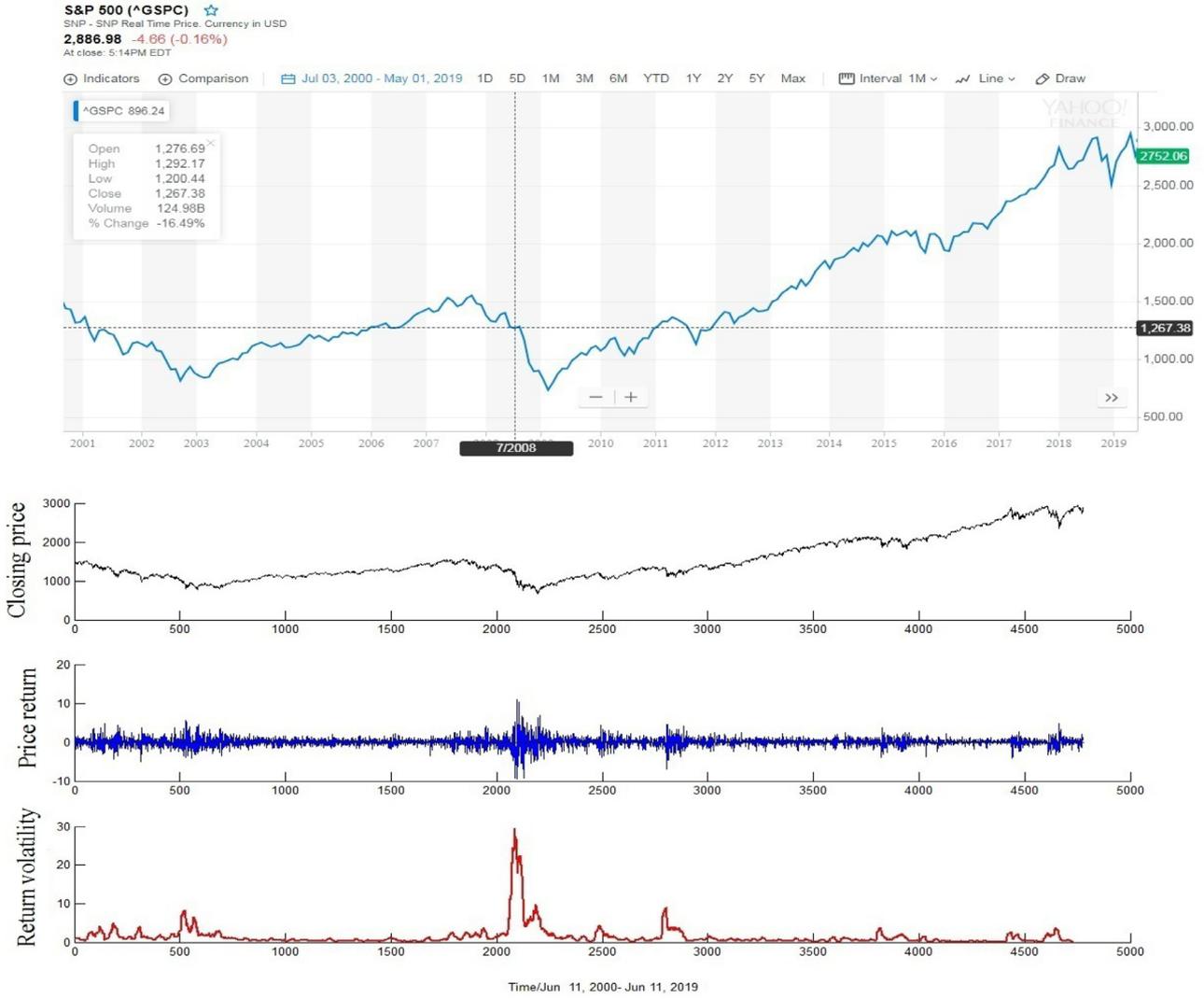


Fig. 4. The chart of daily index market time series S&P 500 from 2000 to 2019.

time series in j levels. Therefore, the HV time series is decomposed into several subseries, a smooth approximate HV time series $g_1(t)$, and detail time series $h_i(t)$, $i = 1, 2, \dots, j$ with smaller wave amplitude. In general, by using BSd multiresolution, the original HV time series $s(t)$ can be reconstructed by the equal relationship of (10). Therefore, all decomposed components are simultaneously used in the relations (11) and (15) as the new inputs of the RNN.

Fig. 5 demonstrates the S&P 500 HV time series $s(t) = \sigma^2(t)$, $t \in [t_1, t_{32}]$, its local decomposition by using BSd, $d = 3$ (BS3), and another of order $d = 1$ (BS1, or Haar) of the $j = 2$ levels of the decomposition. It is easily seen that the smooth approximation HV time series $g_1(t)$ obtained by BS3 with a lower frequency is fitter than the ones obtained by BS1. The smooth detail price time series $h_1(t)$ obtained by BS3 has fewer fluctuations with smaller wave amplitude than the ones obtained by BS1.

3.3. Application of the BSd-RNN model for predicting of volatility time series

According to (21)–(23) and affecting factors on GARCH(3,1) formula, we update the definition for input x_n in relation (12) by:

$$x_n = (\sigma_{n-3}^2, \sigma_{n-2}^2, \sigma_{n-1}^2, \eta_{n-1}, r_{n-1}, \mu_{n-1}, \hat{\sigma}_{n-1}^2) \quad (27)$$

and target output z_n in relation (13) is changed by:

$$z_n = \sigma_n^2 \quad (28)$$

The network output $y_n = \hat{\sigma}_n^2$ is obtained for predicting HV time series at one ($l = 1$) step ahead by using MRA on HV time series σ_t^2 . The smooth approximation time series $g_1(t)$, and detail time series $h_i(t)$, $i = 1, \dots, j$ are also obtained (we can suppose $j = 2$). In addition, according to relations (10) and (27), a new relationship between every input x_n^{MRA} and y_n is prepared by:

$$x_n^{MRA} = (g_{1,n-3}, h_{1,n-3}, h_{2,n-3}, \dots, g_{1,n-1}, h_{1,n-1}, h_{2,n-1}, \eta_{n-1}, r_{n-1}, \mu_{n-1}, \hat{\sigma}_{n-1}^2) \quad (29)$$

and the output y_n in the DWT-RNN model is:

$$m_n = f(Wx_n^{MRA} + Um_{n-1} + b_m) \quad (30)$$

$$y_n = f(Vm_n + b_y) \quad (31)$$

where $f(\cdot)$ is an activation function such as $\tanh(\cdot)$. Also, W , U , V and b are parameters matrices and vector.

Moreover, we prepare a similar definition with (27) for FFNN input x_n by:

$$x_n = (\sigma_{n-3}^2, \sigma_{n-2}^2, \sigma_{n-1}^2, \eta_{n-1}, r_{n-1}, \mu_{n-1}) \quad (32)$$

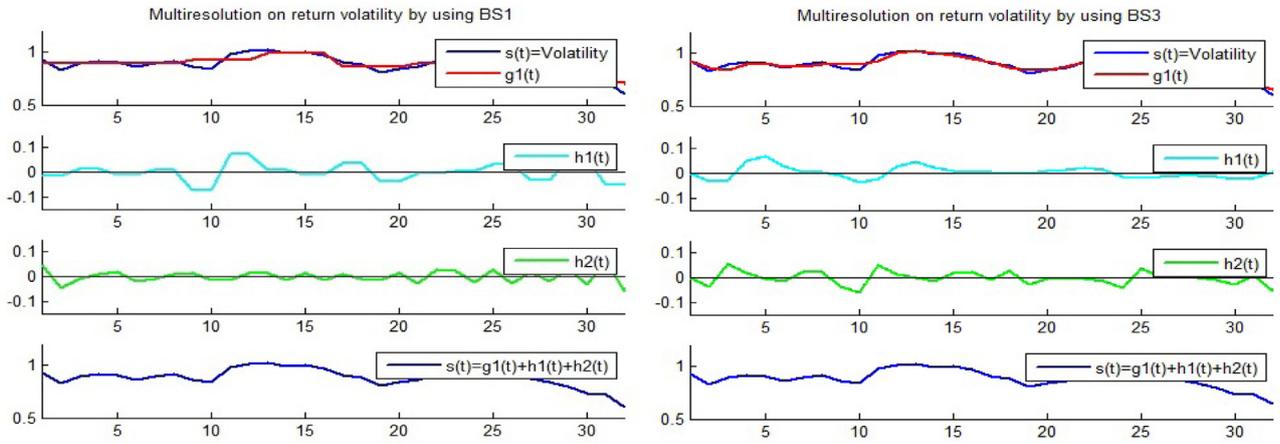


Fig. 5. The S&P 500 volatility multiresolution by using Haar, and BS3 of the $j = 2$ levels of decomposition.

According to relations (10) and (32), the relationship between every new input x_n^{MRA} and y_n is prepared by:

$$x_n^{MRA} = (g_{1,n-3}, h_{1,n-3}, h_{2,n-3}, \dots, g_{1,n-1}, h_{1,n-1}, h_{2,n-1}, \eta_{n-1}, r_{n-1}, \mu_{n-1}) \quad (33)$$

and the output y_n in the DWT-FFNN model with two hidden layers ($M = 2$) is of the following form:

$$y_n = f(W^{(3)} \cdot f(W^{(2)} \cdot f(W^{(1)} \cdot x_n^{MRA} + b_1) + b_2) + b_3) \quad (34)$$

3.4. Experimental forecasting results of the Bsd-RNN model

In order to demonstrate the effectiveness of the BSd wavelets of different orders ($d = 1, 2, 3, 4$) on RNN action, the indices of the S&P 500, NASDAQ, DJIA, and NYSE. were predicted by BS1-RNN, BS2-RNN, BS3-RNN, BS4-RNN, and RNN models, and the prediction test results were compared by performance metrics. The results are tabulated in Table 1. As it is shown, the ranking of predictive ability for BSd-RNN models is almost all superior to the RNN model. It means that the proposed model is able to warrant the accuracy of the RNN by using the BSd. It is shown in Table 1, among all BSd-RNN models, BS3-RNN and then BS4-RNN have better performance than the others.

Table 1 Performance metric of BSd-RNN models on different markets volatility forecasting at one steps ahead.

Index market	Model	Forecasting performance measures		
		FRMSE	FMAE	FMAPE
S&P 500	BS3-RNN	0.0022	0.0012	0.0231
	BS4-RNN	0.0117	0.0292	0.0524
	BS2-RNN	0.0187	0.0314	0.0632
	BS1-RNN	0.0276	0.0354	0.0869
	RNN	0.0306	0.0503	0.0903
	BS3-RNN	0.0064	0.0053	0.0106
NASDAQ	BS4-RNN	0.0105	0.0079	0.0123
	BS2-RNN	0.0116	0.0087	0.0135
	BS1-RNN	0.0183	0.0161	0.0154
	RNN	0.0361	0.0274	0.0406
	BS3-RNN	0.0082	0.0060	0.0104
	BS4-RNN	0.0112	0.0062	0.0107
DJIA	BS2-RNN	0.0115	0.0067	0.0115
	BS1-RNN	0.0147	0.0100	0.0336
	RNN	0.0269	0.0177	0.0392
	BS3-RNN	0.0037	0.0046	0.0101
	BS4-RNN	0.0109	0.0085	0.0143
	BS2-RNN	0.0112	0.0187	0.0202
NYSE	BS1-RNN	0.0245	0.0193	0.0440
	RNN	0.0312	0.0245	0.0694

4. Comparison results of the proposed BSd-RNN model with the other common DWT-ANN models

In time series modelling and forecasting, various types of wavelets such as Haar, Morlet (Li et al., 2016), and Daubechies wavelets can be used. Haar WT has been extensively applied together with the ANN model (Chandar et al., 2016; Jothimani, Shankar, & Yadav, 2015). Nevertheless, because of its discontinuity, it does not approximate high-frequency time series very well (Lahmiri, 2014). dbn is popular because it is compactly supported orthonormal wavelets and provide accurate time series predictions. For example, db3, and db4 are widely used in time series forecasting problems (Bento et al., 2018; Homayouni & Amiri, 2011; Lahmiri, 2014).

In order to demonstrate the more effectiveness of BS3 in the performance of the RNN model, the predicting results of the proposed BS3-RNN model was compared with the similar DWT-RNN

Table 2 Performance metric of several models on different markets volatility forecasting at one steps ahead.

Index market	Model	Forecasting performance measures		
		FRMSE	FMAE	FMAPE
S&P 500	BS3-RNN	0.0022	0.0012	0.0231
	db4 -RNN	0.0126	0.0249	0.0643
	db3-RNN	0.0223	0.0314	0.0859
	Haar-RNN	0.0276	0.0354	0.0869
	BS3-FFNN	0.1242	0.1491	0.2565
	GARCH	0.3578	0.3907	0.4356
NASDAQ	ARIMA	0.4956	0.4091	0.5793
	BS3-RNN	0.0064	0.0053	0.0106
	db4 -RNN	0.0141	0.0116	0.0117
	db3-RNN	0.0174	0.0146	0.0148
	Haar-RNN	0.0183	0.0161	0.0154
	BS3-FFNN	0.1097	0.1465	0.2114
DJIA	GARCH	0.3346	0.2542	0.3434
	ARIMA	0.3304	0.3555	0.3645
	BS3-RNN	0.0082	0.0060	0.0104
	db4 -RNN	0.0132	0.0103	0.0230
	db3-RNN	0.0189	0.0121	0.0284
	Haar-RNN	0.0147	0.0100	0.0336
NYSE	BS3-FFNN	0.1061	0.1313	0.1503
	GARCH	0.2237	0.2044	0.3497
	ARIMA	0.2589	0.2874	0.3909
	BS3-RNN	0.0037	0.0046	0.0101
	db4 -RNN	0.0162	0.0126	0.0361
	db3-RNN	0.0179	0.0138	0.0432
NYSE	Haar-RNN	0.0245	0.0193	0.0440
	BS3-FFNN	0.1169	0.2881	0.2088
	GARCH	0.2825	0.3272	0.2495
	ARIMA	0.3357	0.3123	0.2675

models such as Haar-RNN, db3-RNN, and db4-RNN. The comparison results for forecasting of four indices are tabulated in Table 2. As it is shown, the ranking of predictive ability for BS3-RNN models is almost always superior to the other models which use other wavelets. Generally, for all markets, the performance of BS3-RNN, db4-RNN, db3-RNN, and Haar-RNN models are better than the others, respectively. The preprocessing of network input samples by the continuous wavelets B-Spline of high order (BS3 and BS4) are more appropriate than the discontinuous wavelet (Haar). Also, the preprocessing of network input samples by BS3 as finitely differentiable wavelet is more appropriate than the dbn $n = 3, 4$ (db3, db4), while db4 is differentiable. Moreover, the RNN model had better performance than FFNN for forecasting of high-frequency time series.

The forecasting result errors of non-parametric BS3-RNN model and parametric ARIMA and GARCH models are prepared in Table 2 as well. As it is shown, The BS3-RNN model is more accurate than other ones. There are considerable differences between the errors of GARCH and ARIMA models in one hand and other models on the other hand. Indeed, the performance of non-parametric and non-linear models is much better than the performance of parametric and linear models.

5. Conclusions

The accuracy of an ANN model for predicting the behavior of a time series in the future can be improved by using various wavelets. In this paper, we selected BSd multiresolution as a preprocessing tool to improve the RNN model, which was presented for the prediction of a high-frequency time series. In this forecasting model, named BSd-RNN, the utilization of BSd was justifiable from two aspects: Firstly, bounded matrices were used in constructing low and high-pass filters. These tridiagonal matrices are the cause of the linear algorithms which increase the efficiency of the model. Then, decomposed smooth time series, which were obtained from the BSd multiresolution were applied as the new inputs of the RNN model.

Based on the practical findings, the proposed BSd-RNN model was an efficient model in forecasting nonlinear time series such as volatility time series. As we had extensively investigated, among of the BSd-RNN ($d = 1, 2, 3, 4$) models, BS3-RNN predicted future volatility better than others. BSd-RNN model also predicted future volatility better than other common DWT-RNN models such as Haar-RNN, and dbn-RNN models. The proposed model had better performance than DWT-FFNN, ARIMA, and GARCH models as well.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Credit authorship contribution statement

Zeinab Hajiabotorabi: Conceptualization, Methodology, Validation, Writing - original draft. **Aliyeh Kazemi:** Writing - original draft, Supervision, Data curation. **Faramarz Famil Samavati:** Supervision, Data curation. **Farid Mohammad Maalek Ghaini:** Supervision, Data curation.

References

- Adelson, E. H., Simoncelli, E., & Hingorani, R. (1987). Orthogonal pyramid transforms for image coding. *Visual Communications and Image processing II*, 845, 50–59.
- Alderson, T., Mahdavi-Amiri, A., & Samavati, F. F. (2018). Offsetting spherical curves in vector and raster form. *The Visual Computer*, 34(6–8), 973–984.
- Alderson, T., & Samavati, F. (2014). Optimizing line-of-sight using simplified regular terrains. *The Visual Computer*, 31(4), 407–421.
- Altunkaynak, A., & Ozger, M. (2016). Comparison of discrete and continuous wavelet-multilayer perceptron methods for daily precipitation. *Journal of Hydrologic Engineering*, 21(7), 04016014.
- Bahrammirzaee, A. (2010). A comparative survey of artificial intelligence applications in finance: Artificial neural networks, expert system and hybrid intelligent systems. *Neural Computing and Applications*, 19(8), 1165–1195.
- Bartels, R. H., & Samavati, F. F. (2000). Reversing subdivision rules: local linear conditions and observations on inner products. *Journal of Computational and Applied Mathematics*, 119, 21–67.
- Bartels, R. H., & Samavati, F. F. (2011). Multiresolutions numerically from subdivisions. *Computers and Graphics*, 35(2), 185–197.
- Bento, P. M. R., Pombo, J. A. N., Calado, M. R. A., & Mariano, S. J. P. S. (2018). A bat optimized neural network and wavelet transform approach for short-term price forecasting. *Applied Energy*, 210, 88–97.
- Bildirici, M., & Ersin, O. (2013). Forecasting oil prices: Smooth transition and neural network augmented GARCH family models. *Journal of Petroleum Science and Engineering*, 109, 230–240.
- Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3), 307–327.
- Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: Forecasting and control*. John Wiley & Sons.
- Brunn, M., Sousa, M. C., & Samavati, F. F. (2007). Capturing and reusing artistic styles with reverse subdivision based multiresolution methods. *International Journal of Image and Graphics*, 7(4), 593–615.
- Chandar, S. K., Sumathi, M., & Sivanandam, S. N. (2016). Prediction of stock market price using hybrid of wavelet transform and artificial neural network. *Indian Journal of Science and Technology*, 9(8), 1–5.
- Conejo, A. J., Plazas, M. A., Espinola, R., & Molina, A. B. (2005). Day-ahead electricity price forecasting using the wavelet transform and ARIMA models. *IEEE Transactions on Power Systems*, 20(2), 1035–1042.
- Daubechies, I. (1992). *Ten lectures on wavelets*. Philadelphia: Society for Industrial and Applied mathematics.
- Donaldson, G., & Kamstra, M. (1997). An artificial neural network-GARCH model for international stock return volatility. *Journal of Empirical Finance*, 4(1), 17–46.
- Doucoure, B., Agbossou, K., & Cardenas, A. (2016). Time series prediction using artificial wavelet neural network and multi-resolution analysis: Application to wind speed data. *Renewable Energy*, 92, 202–211.
- Elman, J. L. (1990). Finding structure in time. *Cognitive science*, 14(2), 179–211.
- Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica: Journal of the Econometric Society*, 50, 987–1008.
- Finkelstein, A., & Salesin, D. H. (1994). Multiresolution curves of SIGGRAPH. *Computer graphics, annual conference series*.
- Grossmann, A., & Morlet, J. (1984). Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM Journal on Mathematical Analysis*, 15(4), 723–736.
- Haar, A. (1985). *Results in mathematics*. Springer. 8 (2), 194–196
- Homayouni, N., & Amiri, A. (2011). Stock price prediction using a fusion model of wavelet, fuzzy logic, and ANN. In *International conference on e-business, management and economics*: 25 (pp. 277–281). Singapore
- Hsieh, T. J., Hsiao, H. F., & Yeh, W. C. (2011). Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm. *Applied soft computing*, 11(2), 2510–2525.
- Huang, L., & Wang, J. (2018). Forecasting energy fluctuation model by wavelet decomposition and stochastic recurrent wavelet neural network. *Neurocomputing*.
- Jothimani, D., Shankar, R., & Yadav, S. S. (2015). Discrete wavelet transform-based prediction of stock index: A study on national stock exchange fifty index. *Journal of Financial Management and Analysis*, 28(2), 35–49.
- Kazemi, A., Shakouri, H. G., Menhaj, M. B., Mehregan, M. R., & Neshat, N. A. (2010). Hierarchical artificial neural network for transport energy demand forecast: Iran case study. *Neural Network World*, 20(6), 761.
- Khuat, T. T., Le, Q. C., Nguyen, B. I., & Le, M. H. (2016). Forecasting stock price using wavelet neural network optimized by directed artificial bee colony algorithm. *Journal of Telecommunications and Information Technology*, 43–52.
- Kim, H. Y., & Won, C. H. (2018). Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. *Expert Systems with Applications*, 103, 25–37.
- Krishnaprasad, S., & Pati, C. (1993). Analysis and synthesis of feed-forward neural networks using discrete affine wavelet transformations. *IEEE Transactions on Neural Networks*, 4(1), 73–85.
- Kristjanpoller, W., & Hernandez, E. (2017). Volatility of main metals forecasted by a hybrid ANN-GARCH model with regressors. *Expert Systems with Applications*, 84, 290–300.
- Kristjanpoller, W., & Minutolo, M. C. (2016). Forecasting volatility of oil price using an artificial neural network-GARCH model. *Expert Systems with Applications*, 65, 233–241.
- Lahmiri, S. (2014). Wavelet low and high frequency components as features for predicting stock prices with back propagation neural networks. *Journal of King Saud University Computer and Information Sciences*, 26(2), 218–237.
- Li, Y., Li, X., & Wang, H. (2016). Based on multiple scales forecasting stock price with a hybrid forecasting system. *American Journal of Industrial and Business Management*, 6, 1102–1112.
- Luo, R., Zhang, W., Xu, X., & Wang, J. (2018). A neural stochastic volatility model. In *Thirty-second AAAI conference on artificial intelligence*.

- Mallat, S. (1999). *A wavelet tour of signal processing*. Elsevier.
- Mandal, Haque, Meng, J., Srivastava, & Martinez, R. (2013). A novel hybrid approach using wavelet, firefly algorithm, and fuzzy ARTMAP for day-ahead electricity price forecasting. *IEEE Transactions on Power Systems*, 28(2), 1041–1051.
- Meyer, Y. (1992). *Wavelets and applications*: 31. Paris: Masson.
- Mishra, N., Soni, H. K., & Sharma, S. (2018). Upadhyay a.k. development and analysis of artificial neural network models for rainfall prediction by using time-series data. *International Journal of Intelligent Systems and Applications*, 10(1), 16–23.
- Moller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6(4), 525–533.
- Moltaji, A., Runions, A., & Samavati, F. F. (2017). Subdivision and multiresolution for pups. *Computers & Graphics*, 62, 53–66.
- Nguyen, T., & He, T. X. (2015). Wavelet analysis and applications in economics and finance. *Research & Reviews: Journal of Statistics and Mathematical Sciences*, 1(1), 22–37.
- Olsen, L., & Samavati, F. F. (2008). A discrete approach to multiresolution curves and surfaces. In *International conference on computational science and its applications (ICCSA)* (pp. 468–477). IEEE Computer Society.
- Qu, Z., Mao, W., Zhang, K., Zhang, W., & Li, Z. (2019). Multi-step wind speed forecasting based on a hybrid decomposition technique and an improved back-propagation neural network. *Renewable Energy*, 133, 919–929.
- Samavati, F. F., Mahdavi-Amiri, N., & Bartels, R. H. (2002). Multiresolution representation of surface with arbitrary topology by reversing do subdivision. *Computer Graphic Forum*, 21(2), 121–136.
- Samavati, F. F., & Bartels, R. H. (1999). Multiresolution curve and surface representation by reversing subdivision rules. *Computer Graphics Forum*, 18(2), 97–119.
- Samavati, F. F., Bartels, R. H., & Olsen, L. (2007). Local b-spline multiresolution with examples in iris synthesis and volumetric rendering, image pattern recognition: Synthesis and analysis in biometrics. *Series in Machine Perception and Artificial Intelligence*, 65–101.
- Shaghaghí, S., Bonakdari, H., Gholami, A., Ebtehaj, I., & Zeinolabedini, M. (2017). Comparative analysis of GMDH neural network based on genetic algorithm and particle swarm optimization in stable channel design. *Applied Mathematics and Computation*, 313, 271–286 .
- Shi, B., Wang, P., Jiang, J., & Liu, R. (2018). Applying high-frequency surrogate measurements and a wavelet-ANN model to provide early warnings of rapid surface water quality anomalies. *Science of the Total Environment*, 610, 1390–1399.
- Stromberg, J. (1981). A modified Haar system and higher order spline systems. *Conference on Harmonic Analysis in Honor of Antoni Zygmund*, 475–493.
- Stollnitz, E. J., Deroose, A. D., & Salesin, D. H. (1996). *Wavelets for computer graphics: Theory and applications*. The Morgan Kaufmann Publishers.
- Taha, S. M., & Taha, Z. K. (2018). EEG signals classification based on autoregressive and inherently quantum recurrent neural network. *International Journal of Computer Applications in Technology*, 58(4), 340–351.
- Unser, M. (1997). Ten good reasons for using spline wavelets. In *SPIE conf. wavelet applications in signal and image processing v: vol. 3169* (pp. 422–431). USA
- Vaisla, K. S. (2010). Stock market forecasting using artificial neural network and statistical technique: A comparison report. *International Journal of Computer and Network Security*, 2(8), 50–55.
- Wang, J. Z., Wang, J. J., Zhang, Z. G., & Guo, S. P. (2011). Forecasting stock indices with back propagation neural network. *Expert Systems with Applications*, 38(11), 14346–14355.
- Wang, L., Zou, H., Su, J., Li, L., & Chaudhry, S. (2013). An ARIMA-ANN hybrid model for time series forecasting. *Systems Research and Behavioral Science*, 30(3), 244–259.
- Wecker, L., Samavati, F., & Gavrilova, M. A. (2010). Multiresolution approach to iris synthesis. *Computers and Graphics*, 34(4), 468–478.
- Willmott, J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1), 79–82.
- Xing, F. Z., Cambria, E., & Zhang, Y. (2019). Sentiment-aware volatility forecasting. *Knowledge-Based Systems*, 176, 68–76.
- Zhang, Q., & Benveniste, A. (1992). Wavelet networks. *IEEE Transactions on neural networks*, 3(6), 889–898.