Computers & Geosciences 138 (2020) 104461

Contents lists available at ScienceDirect

Computers and Geosciences

journal homepage: www.elsevier.com/locate/cageo

Research paper

A comparative study of wavelet-based ANN and classical techniques for geophysical time-series forecasting

Shivam Bhardwaj[a], E. Chandrasekhar[b,*], Priyanka Padiyar[c], Vikram M. Gadre[a]

[a] Department of Electrical Engineering, IIT Bombay, Powai, Mumbai 400076, India
[b] Department of Earth Sciences, IIT Bombay, Powai, Mumbai 400076, India
[c] Department of Electronics and Telecommunications Engineering, Goa College of Engineering, Goa 403401, India

ARTICLE INFO

Keywords:
Time-series forecasting
ARIMA models
Kalman filter algorithm
Wavelet transform
Artificial neural networks

ABSTRACT

Time-series modeling forms an important area of research in geophysics. Time-series models can be linear, like linear state-space models or non-linear, like artificial neural networks. One way to judge the goodness of different models associated with a given time-series is to assess the prediction capabilities of these models. Some of the important techniques used in time-series forecasting are: (i) Minimum mean squared error (MMSE) forecast obtained using conditional means of ARIMA(p,d,q) models, (ii) Kalman filter approach and (iii) Artificial neural networks (ANN) approach. However, the wavelet-based versions of these techniques, respectively denoted as W-MMSE, W-Kalman and W-ANN, rather than the original techniques themselves, have been found to possess better capabilities in forecasting the highly nonlinear geophysical data. Using the original and predicted data, these observations have been validated by determining the RMSE (root mean squared error) and correlation coefficients between them. The prediction capabilities of both versions of the above techniques are tested on (i) the ionospheric total electron content (TEC) data, (ii) the daily average rainfall data and (iii) the gamma-ray log data from an offshore oil well, off the west coast of India. The TEC and daily average rainfall data sets designate as examples of data with very high correlations pertaining over very large lags. They also have a strong seasonality component associated with them. However, the gamma-ray log data sets show no seasonality component and have no trends associated with them. Therefore, the choice of different nonlinear data sets having diverse sources of their origins are apt to test the forecasting capabilities of these techniques. It has been observed that W-ANN gives the best prediction, when compared with the other algorithms discussed in this paper. This is believed to be due to the use of non-linear activation function by ANNs to produce regression that results in capturing the inherent non-linear dynamics of the process effectively. The results also show the usefulness of discrete wavelet transform (DWT) coefficients as training features for both linear and non-linear forecasting approaches. The better performance of the wavelet-based forecasting algorithms can be attributed to the fact that DWT coefficients are wide-sense stationary sequences. Thus the wavelet-based versions of these models like W-MMSE and W-Kalman provide better fit to these coefficients, compared to the original time-series data itself.

1. Introduction

A thorough understanding of non-linear geophysical signals is essential to get a deeper insight into the dynamics governing a complex geophysical system generating such signals (Holloway Jr., 1958; Huang et al., 1998; Percival and Walden, 2006). One way to infer the dynamics of the system from a given time-series is to develop plausible statistical models that display high degree of correlation with the signals under study. However, different models with varying complexities can also be fit to a given non-linear time-series. The quantitative assessment of the suitability of the derived statistical model to the data in question essentially depends on the number of free parameters that the model has. In other words, the number of free parameters in any statistical model dictates the efficacy of the model in forecasting the signal with least error. By free parameters we mean, the parameters that are varied to find the optimum fit for a model, once the data is given.

Time-series forecasting is an important area of research, in which past values of the data are used to predict future values by developing a statistical model, which facilitates to develop a statistical framework to predict the future values of the system with least predictable error. This kind of modeling approach is useful, when there is little or no

[*] Corresponding author.
E-mail address: esekhar@iitb.ac.in (E. Chandrasekhar).

https://doi.org/10.1016/j.cageo.2020.104461
Received 15 April 2019; Received in revised form 22 February 2020; Accepted 24 February 2020
Available online 29 February 2020

information available about different variables and relationship that exists between them.

Earlier approaches to time-series forecasting involved fitting linear additive models like auto regressive integrated moving average (ARIMA) model to the given time-series and using it to predict the data (Wold, 1938; Box et al., 2015). These predictions are based on the conditional expectation of ARIMA models and have shown to predict with minimum mean squared error (MMSE) (Hamilton, 1994). That means, if we have a set of finite observations, $X_1, X_2, \dots, X_N$, in the form of a time-series, and if $\hat{X}_{N+1}$ denotes the prediction or forecast of the true future value X_{N+1}, then the MMSE forecast is the one which minimizes the conditional variance of error given by $\mathbf{E}[(X_{N+1} - \hat{X}_{N+1})^2 | X_1, X_2, \dots, X_N]$. Here $\mathbf{E}[.]$ is the expectation operator.

The state-space approach (Kailath, 1980) is a unified framework to deal with different types of problems in time-series analysis. In state-space approach, the relationship between the observations (time-series), $X_t = (X_1, X_2, \dots, X_N)^T$ and the state-vector, $z_t = (z_1, z_2, \dots, z_N)^T$ is governed by the state-space model. The state-space model for a time-series is given by

$$z_{t+1} = \Phi z_t + w_t \tag{1}$$

$$X_t = C z_t + v_t \tag{2}$$

where, $\Phi \in \mathbf{R}^{N\times N}$ is the state-transition matrix relating the current state to future state and $C \in \mathbf{R}^{1\times N}$ is the output matrix relating the current value of the time-series to the current value of the state-variable. w_t and v_t are uncorrelated (in time and with each other) random sequences with zero mean and their respective covariances are symmetric, positive semi-definite matrices Q and R such that $\mathbf{E}[w_t w_t^T] = Q$ and $\mathbf{E}[v_t v_t^T] = R$. The main aim of the state-space analysis is therefore to find out the evolution of the vector, once the observations from the system are given in the form of a time-series (Durbin and Koopman, 2002, 2012). Once the description of time-series under consideration is given in the state-space form, then one can use recursive Kalman filter algorithm (Kalman, 1960) to determine the states z_t of the system, which in turn can be used to forecast future values of the time-series (using Eq. (2)) (Kailath, 1980). Kalman filter is an optimal state-observer that provides the estimate of the state with minimum error co-variance (Kalman, 1960). This holds true if the description of the system under consideration is given in the form of a linear state-space model. So, linearity is an important assumption, while implementing Kalman filter algorithm. Apart from linearity, another important assumption is that the input noise w_t and measurement noise v_t are second order stationary Gaussian sequences.

Artificial neural networks (ANNs) have proven to be very useful in pattern classification and pattern recognition. ANNs are inspired by biological network of neurons in a human brain and are capable of learning patterns from the data. One of the important areas where ANNs are very useful, is the time-series forecasting (Park et al., 1991; Hill et al., 1994, 1996; Zhang et al., 1998). Several factors make ANNs as very important tools to address the forecasting problem. Firstly, while most of the forecasting techniques assume a certain underlying model for the data under study, ANNs provide a data-driven self adaptive approach for time-series forecasting without assuming any apriori model (Hecht-Nielsen, 1992). Secondly, if the number of free parameters in the network are chosen wisely to avoid the problem of overfitting, then the ANNs give quite a good prediction for the test data. Third, ANNs can approximate any continuous functional relation that exists between the input and the output of the network with a high degree of accuracy (Funahashi, 1989; Cybenko, 1989; Hagan and Menhaj, 1994). Finally, forecasting algorithms based on ARIMA models and Kalman filter approach work under the assumption that the system generating the data is linear. Since most geophysical systems have highly non-linear dynamics associated with them and since ANNs use non-linear activation function to predict the output, forecasting algorithms based on ANN can best capture the inherent non-linearity in the system (Specht, 1991; Sarle, 1994). Since the advent of deep learning, several modification to the basic multi layer perceptron (MLP) architectures have been introduced for the purpose of time series analysis (Längkvist et al., 2014; Qiu et al., 2014). Deep learning architectures like RNN (recurrent neural network), LSTM (long–short term memory networks) and GRU (gated recurrent units) are some of the popular modifications in the existing MLP architecture of ANNs, which take advantage of sequential nature of the time-series data (Coulibaly and Baldwin, 2005; Gers et al., 2002; Yao et al., 2017).

In recent years, wavelet analysis has proven to be an important technique in the area of time-series analysis (Percival and Walden, 2006; Daubechies, 1992; Nason and Von Sachs, 1999; Chandrasekhar and Rao, 2012; Sharma et al., 2013; Luo et al., 2016). Discrete wavelet transform (DWT) can be directly applied to a time-series recorded at discrete instances of times. DWT gives a multiscale decompositions of the signal under consideration (Doucoure et al., 2016; Chen et al., 2006). Mallat's algorithm (Mallat, 1989) is widely accepted for numerical implementation of DWT. At each level j of the decomposition, one can get an approximation of the data at j denoted by the sequence a_j, known as approximation coefficients. It is assumed that at scale $j = 0$, $a_0[n] \sim X_n, n \in \{1, 2, \dots, N\}$, where $a_0[n]$ represents the finest approximation of the data. However, the sequence of coefficients a_j, for $j > 0$ represent coarser approximation of the data. The information lost between any two consecutive approximations of the data is represented by a set of detail coefficients d_j. Accordingly, for $j > 0$ we have;

$$a_{j+1}[n] = \sum_{k=-\infty}^{\infty} h[2n-k] a_j[k] \tag{3}$$

$$d_{j+1}[n] = \sum_{k=-\infty}^{\infty} g[2n-k] a_j[k] \tag{4}$$

Here, $h[n]$ and $g[n]$ are quadrature mirror filters (QMF) related to each other via $g[n] = (-1)^n h[1-n]$ (Mallat, 1989; Daubechies, 1992). These coefficients are the optimum features extracted from the signal at different decomposition levels and can be used as new features to train a multilayer perceptron or ANN for time-series prediction (Yousefi et al., 2005; Partal and Cigizoglu, 2009; Stolojescu et al., 2010).

In the present study, we show that if the data forecasting techniques like MMSE of ARIMA, Kalman filter approach and ANN are used together with wavelet analysis, then the performance of the prediction models improves drastically, compared to the case, when these techniques are directly implemented on the data for prediction. In other words, the wavelet-based versions of the above techniques largely improve the prediction accuracy of the time-series data. The metrics used for comparison of the forecast performance are root mean squared (RMS) error and correlation coefficient.

The organization of the paper is as follows: Section 2 explains the database used for the performance analysis of the aforementioned forecasting algorithms. Section 3 describes various forecasting algorithms and their wavelet-based formalism. Section 4 discusses the results of different forecasting algorithms in the prediction of daily average rainfall data, TEC data and geophysical well-log data. Finally, Section 5 provides the conclusions of this study.

2. Data

For the present study, GPS data corresponding to the solar minimum (2008) and solar maximum (2014) years have been obtained from Scripps Orbit and Permanent Array Centre (SOPAC), California (see http://sopac.ucsd.edu/dataBrowser.shtml), where a repository of GPS data from different IGS (International GNSS Service) stations (see http://www.igs.org/network) is maintained. Full description of the downloaded data and the procedure to calculate TEC values from GPS data can be found in Klobuchar et al. (1996), Seemala and Valladares (2011) and Chandrasekhar et al. (2016) (see also http://seemala.blogspot.in/). TEC data from six different stations was collected. Every cycle (per day)

Table 1
Description of different data-sets used for analysis in the current manuscript.

Data-set	Number of features	Number of samples in training data	Number of samples in test data
TEC data (all stations)	1	14 400	1440
Average rainfall data (all stations)	1	2920	365
Well-log data	1	1950	50

of TEC data has 1440 points (with a sampling interval of one minute). Data of past 10 days was used to predict the data points of 11th day.

Secondly, five different sets of daily average rainfall data were collected from various geographical locations of India such as Goa, Mumbai, Jaipur, New Delhi and Cherrapunji, having different rainfall patterns. Future prediction for the year 2017 was done by using datasets from 2009–2016 for different forecasting algorithms. Also, gamma-ray log data from an offshore well of ONGC (Oil and Natural Gas Corporation), India, was used to test the performance of the forecasting algorithm. Table 1 gives a brief description of the time-series data considered for analysis in the present manuscript.

3. Methodology

This section briefly describes different forecasting algorithms used in the current study. In the first approach, some part of the observations (known as training data) were used to train (i) ARIMA(p,d,q) models (ii) Kalman filter and (iii) an ANN to predict the future values. These values were then compared with the true values (referred to as the test data) for calculating RMSE and correlation coefficient. In the second approach, the wavelet-based version of the above techniques, namely, (i) W-MMSE of ARIMA(p,d,q) models, (ii) W-Kalman and (iii) W-ANN were introduced, wherein the ARIMA(p,d,q) models, Kalman filter and ANN were trained using the wavelet coefficients of the training data to predict the wavelet coefficient of the test data. Once the wavelet coefficients were predicted, then the inverse wavelet transform (Sharma et al., 2013; Mallat, 1989) was performed to reconstruct the time-series. The reconstructed time-series was then compared with the test data to compute the RMSE and correlation coefficients. Detailed description of these two approaches is given below.

(i) Let $X_1, X_2, \dots, X_N$ denote a set of observations in the form of time-series corresponding to a geophysical process. Then, a forecast of the process at a forecast horizon h is denoted as $\hat{X}_{N+h}$ and can be thought of as an outcome of a functional relation relating $\hat{X}_{N+h}$ to the past forecasts $\hat{X}_{N+1}, \hat{X}_{N+2}, \dots, \hat{X}_{N+(h-1)}$ and to the given data, $X_1, X_2, \dots, X_N$ i.e., $\hat{X}_{N+h} = f(X_1, X_2, \dots, X_N, \hat{X}_{N+1}, \hat{X}_{N+2}, \dots, \hat{X}_{N+(h-1)})$. Depending upon the model imposed on the data, the function $f(.)$ can be either linear or non-linear. Parametric approaches like MMSE forecast using ARIMA(p,d,q) models and Kalman filter method assume that there is an underlying linear model, which is responsible for generating the observed data. However, ANN does not assume any such constraint on the model to generate the data. In the first approach, different forecasting algorithms were applied to the data directly for prediction and their individual performances were assessed subsequently. Next, the results of individual techniques were compared using their respective metrics like root mean-squared error (RMSE) and correlation coefficient.

(ii) In the second approach, wavelet-based formulation for the above techniques were developed for forecasting the data. Given the data $X_n, n \in \{1, 2, \dots, N\}$ one can obtain the wavelet decomposition of the data. This wavelet decomposition is characterized by the scaling and detail coefficients, a_j and d_j, respectively. If we have the knowledge of scaling and detail coefficients at the scale j_0, then one can re-construct the original data using the inverse discrete wavelet transform (IDWT),

$$X_n = \frac{1}{\sqrt{N}} \sum_k a_{j_0}[k]\phi_{j_0,k}[n] + \frac{1}{\sqrt{N}} \sum_{j \le j_0} \sum_k d_j[k]\psi_{j,k}[n] \tag{5}$$

Here $a_{j_0}[k]$ and $d_j[k]$ were obtained using Eqs. (3) and (4) respectively are called as the approximation coefficients at the scale j_0 and detail coefficients at the scale $j \le j_0$. Also, $\phi_{j_0,k}$ and $\psi_{j,k}$ are discrete scaling and wavelet functions defined on the interval $[1, M]$, $M \in \mathbf{R}$ such that, $\phi_{j_0,k} = 2^{\frac{j_0}{2}}\phi(2^{j_0}n - k)$ and $\psi_{j,k} = 2^{\frac{j}{2}}\psi(2^j n - k)$. Scaling and wavelet functions are generated using the quadrature mirror filter (QMF) $g[k]$ and $h[k]$ and using the dilation equation

$$\phi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} h[k]\phi(2t - k) \tag{6}$$

$$\psi(t) = \sqrt{2} \sum_{k=-\infty}^{\infty} g[k]\phi(2t - k) \tag{7}$$

Now, it can be noticed that at every resolution the number of wavelet coefficients decreases to half of the number of coefficients at finer resolution. Once a data is given, the task is to identify the best combination of the wavelet and scaling functions that best describe the data. Hence, different combinations of filters or different wavelet decompositions are tested on the data to check which wavelet function together with its respective scaling function gives the best reconstruction of the data in terms of error. Also, the level of decomposition is an important parameter that might affect the performance of the reconstruction. It is noticed that as the level of decomposition increases, the error in the reconstruction decreases. Also, this decrease in reconstruction error assumes a constant value after a certain level of decomposition, if this level is denoted by j_0, then, the wavelet decomposition is performed up to this level. Now, if one requires to predict h step ahead forecast then this translates to predicting $\left\lceil \frac{h}{2} \right\rceil$ coefficients at level 1, $\left\lceil \frac{h}{4} \right\rceil$ coefficients at level 2 and finally $\left\lceil \frac{h}{2^{j_0}} \right\rceil$ coefficients at level j_0, where $\lceil (.) \rceil$ is the ceiling function. Then Eq. (5) is used to reconstruct the time-series. This reduction in the number of coefficients as discussed above holds true for univariate time-series data. In the case of multivariate time series, the reduction in the number of coefficients might not follow this pattern.

The underlying theory behind these forecasting algorithms is briefly discussed below. For different methods, the results obtained by forecasting using direct time-series values are compared with the forecast obtained by wavelet transform-based approaches.

3.1. MMSE and W-MMSE time-series forecasting using ARIMA(p,d,q) models

To generate the forecast using conditional mean models, firstly an ARIMA (p,d,q) model was fitted to the data and then using past p values of the process and past q values of the innovation sequence, ϵ_t, the forecasts were generated iteratively. If X_t is represented by an ARIMA(p,d,q) model, then

$$\phi(B)\nabla^d X_t = \theta_0 + \theta(B)\epsilon_t \tag{8}$$

Here, B is a backward shift operator, that is, $BX_t = X_{t-1}$ and ∇ is the difference operator given by $\nabla = 1 - B$. Also,

$$\phi(B) = 1 + \phi_1 B + \phi_2 B^2 + \cdots + \phi_p B^p \tag{9}$$

$$\theta(B) = \theta_1 B + \theta_2 B^2 + \cdots + \theta_q B^q \tag{10}$$

d is the integer order differencing and $\epsilon_t \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$. Estimation of the parameters of an ARIMA (p,d,q) process is done in two steps:

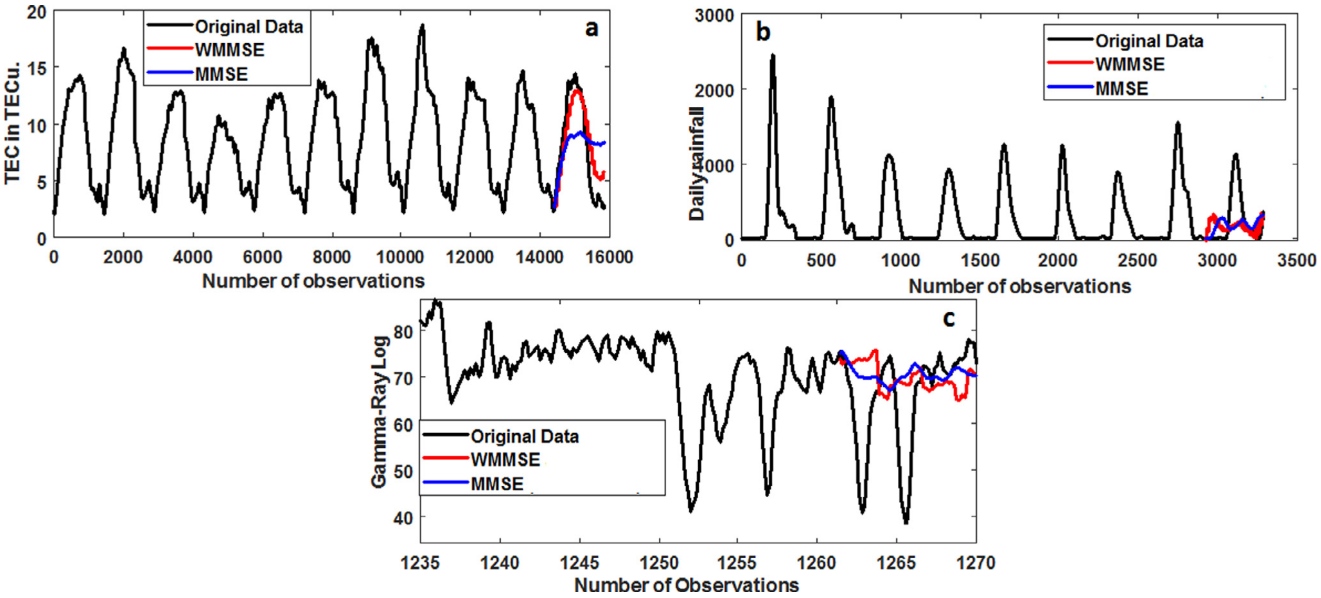

Fig. 1. Results of MMSE and W-MMSE forecast for (a) TEC data for ALGO-2008 (b) rainfall data for Mumbai (c) geophysical well-log data.

- Given the observations, X_t, we assume that the observations are derived from the ARIMA(p,d,q) process as described in Eq. (8). Under this assumption, the differencing of X_t is carried out d times to make it stationary autoregressive moving average (ARMA) process Z_t

  $$\phi(B)Z_t = \theta(B)\epsilon_t + \theta_0$$

  where, $Z_t = \nabla^d X_t$.
- Use conditional likelihood maximization (Box et al., 2015; Hamilton, 1994) to estimate the parameters of the ARMA model.

Once the parameters are estimated, the forecasts are then generated iteratively. For example consider ARIMA(2,0,0) process

$$X_t = C - \phi_1 X_{t-1} - \phi_2 X_{t-2} - \epsilon_t$$

the MMSE forecasts are generated as

$$\begin{aligned}\hat{X}_{N+1} &= C - \phi_1 X_N - \phi_2 X_{N-1}\\ \hat{X}_{N+2} &= C - \phi_1 \hat{X}_{N+1} - \phi_2 X_N\\ \hat{X}_{N+3} &= C - \phi_1 \hat{X}_{N+2} - \phi_2 \hat{X}_{N+1}\\ &\vdots\end{aligned}$$

For a wide sense stationary AR(2) process, this recursion converges to the unconditional mean of the process (Box et al., 2015)

$$\mu = \frac{C}{1 - \phi_1 - \phi_2}$$

In W-MMSE, the DWT coefficients of the training data are used to train an ARIMA(p,d,q) model. Then the estimated model is used to predict the DWT coefficients of the test data using the procedure described above. Once the DWT coefficients are predicted, then the inverse wavelet transform (Eq. (5)) is used to reconstruct the time-series. Fig. 1 and Tables 2–4 show the results of the forecasting of different geophysical data sets based on MMSE and W-MMSE techniques. It is evident from this figure that W-MMSE produces better forecast than ordinary MMSE.

MMSE forecasting minimizes the $\mathbf{E}[(\hat{X}_{N+h} - X_{N+h})^2]$, where $\hat{X}_{N+h}$ is the predicted value of the process at h^{th} instant. These forecast values are generated by finding the parameters of an appropriate ARIMA (p,d,q) process that fits the given data. Maximum likelihood-based parameter estimation (Hamilton, 1994) can be used to obtain the parameters of the ARIMA model. The value of the likelihood function indicates the goodness of the fit of the ARIMA model to a given data. To address the issue of goodness of the fit to the data, one can use Akaike information criteria (AIC) (Akaike, 1969) or Bayesian information criteria (BIC) (Posada and Buckley, 2004). If N designates the length of the data, $\hat{\mathcal{L}}$, the value of the likelihood function and k, the number of free parameters in the model, then the AIC and BIC are respectively given by

$$AIC = 2k - 2\log\hat{\mathcal{L}} \tag{11}$$

$$BIC = (\log N)k - 2\log\hat{\mathcal{L}} \tag{12}$$

The forecast is generated from the model that has the minimum AIC or BIC. Fig. 6 shows the result of forecasting of TEC data corresponding to the site ALGO in the year 2008. Note that when the MMSE forecasts were generated based on direct time-series values, then different models were selected by AIC and BIC and hence different forecasts were generated. However, in the case of W-MMSE, when ARIMA models were trained using the DWT coefficients of training data to produce the DWT coefficients of test data, then, both AIC and BIC selected the same model and gave the same predictions. In general, for a given ARIMA(p, d, q) model the number of free parameters, k, is given by $k = p + q + 2$. Here, p is the number of AR coefficients q is the number of MA coefficients. The variance of innovation sequence (ϵ_t) (Eq. (8)) along with an additive constant was estimated using the maximum likelihood based parameter estimation. For generating the MMSE forecast in Fig. 1a–c the values of p and q were varied between (1–3) and (0–3) respectively while the values of the differencing parameter d was varied between (0–2). The value of the differencing parameter was 0 in all the cases. This is because, the non-zero value of the differencing parameter indicates the presence of increasing or decreasing trends in the data. Since all the data sets considered in the present work contain no trends whatsoever, the ARIMA(p,d,q) models with $d = 0$ are used for forecasting.

3.2. Kalman filter and W-Kalman algorithms for time-series forecasting

Kalman filters assume that the process under analysis has a state-space model (see Eqs. (1), (2)). Given the knowledge of initial state (z_0), the measurement of input (w_t) and the output (X_t), they generate the prediction of the states $\hat{z}_t$, such that the trace of the error covariance matrix $P = \mathbf{E}[(z_t - \hat{z}_t)(z_t - \hat{z}_t)^T]$ is minimized. Hence, the variance associated with each component in the error vector $e_t = (z_t - \hat{z}_t)$ is

Table 2
Prediction results, RMS error and correlation coefficients for TEC data.

	ALGO 2008	ALGO 2014	BRAZ 2008	BRAZ 2014	QIKI 2008	KELY 2014
MMSE (RMS err.)	0.2245(AIC)	0.3485(AIC)	0.1962(AIC)	0.2029(AIC)	0.1805(AIC)	0.1759(AIC)
	0.2344(BIC)	0.3845(BIC)	0.1962(BIC)	0.2328(BIC)	0.1805(BIC)	0.1759(BIC)
Corr. Coeff.	0.8976(AIC)	0.9761(AIC)	0.8909(AIC)	0.9071(AIC)	0.8929(AIC)	0.9304(AIC)
	0.8879(BIC)	0.9461(BIC)	0.8909(BIC)	0.8783(BIC)	0.8929(BIC)	0.9304(BIC)
W-MMSE (RMS err.)	**0.1076**	**0.1113**	**0.1083**	**0.1097**	**0.1065**	**0.1261**
Corr. Coeff.	**0.9762**	**0.9656**	**0.9791**	**0.8960**	**0.9943**	**0.8800**
Kalman (RMS err.)	0.1885	0.1554	0.1423	0.2005	0.0993	0.1062
Corr. Coeff.	0.8686	0.9808	0.9966	0.9690	0.9961	09576
W-Kalman (RMS err.)	**0.1612**	**0.0816**	**0.0941**	**0.1096**	**0.1043**	**0.1343**
Corr. Coeff.	**0.9856**	**0.9434**	**0.9561**	**0.8962**	**0.9943**	**0.8913**
ANN	0.0521	0.1088	0.0997	0.1227	0.0985	0.1278
Corr. Coeff.	0.9747	0.9426	0.9867	0.9042	0.9438	0.8800
W-ANN (RMS err.)	**0.0204**	**0.0796**	**0.0921**	**0.1127**	**0.0955**	**0.1259**
Corr. Coeff.	**0.9980**	**0.9666**	**0.9971**	**0.8906**	**0.9943**	**0.8800**

Table 3
Prediction results, RMS error and correlation coefficients for rainfall data.

	Cherapunji	Goa	Jaipur	Mumbai	Delhi
MMSE (RMS err.)	0.2229(AIC)	0.3110(AIC)	0.1464(AIC)	0.1474(AIC)	0.0973(AIC)
	0.2229(BIC)	0.3110(BIC)	0.1542(BIC)	0.1474(BIC)	0.0961(BIC)
Corr. Coeff	0.7456(AIC)	0.3241(AIC)	0.4188(AIC)	0.5168(AIC)	0.6488(AIC)
	0.7456(BIC)	0.3241(BIC)	0.4050(BIC)	0.5168(BIC)	0.5561(BIC)
W-MMSE (RMS err.)	**0.1669**	**0.1243**	**0.1134**	**0.0742**	**0.0932**
Corr. Coeff.	**0.7074**	**0.9358**	**0.8976**	**0.9316**	**0.8135**
Kalman (RMS err.)	0.3343	0.2969	0.1336	0.1714	0.0924
Corr. Coeff.	0.9155	0.9431	0.9067	0.6622	0.7434
W-Kalman (RMS err.)	**0.1589**	**0.1163**	**0.1034**	**0.0724**	**0.0911**
Corr. Coeff.	**0.7744**	**0.9538**	**0.8766**	**0.9136**	**0.8315**
ANN (RMS err.)	0.1599	0.0700	0.1216	0.0715	0.1267
Corr. Coeff.	0.7604	0.9588	0.8639	0.9162	0.5638
W-ANN (RMS err.)	**0.1569**	**0.0535**	**0.1805**	**0.0542**	**0.0901**
Corr. Coeff.	**0.7704**	**0.9708**	**0.8896**	**0.9256**	**0.8085**

Table 4
RMS error and correlation coefficient for well-log data.

	Well-log
MMSE (RMS err.)	0.1412(AIC)
	0.1382(BIC)
Corr.Coeff.	0.9910(AIC)
	0.9912(BIC)
W-MMSE (RMS err.)	**0.1400**
Corr.Coeff.	**0.9931**
Kalman (RMS err.)	0.3866
Corr.Coeff.	0.9758
W-Kalman (RMS err.)	**0.1284**
Corr. Coeff.	**0.9946**
ANN (RMS err.)	0.1038
Corr. Coeff.	0.9919
W-ANN (RMS err.)	**0.0972**
Corr. Coeff	**0.9906**

minimized. Given the observations up to time $t-1$, if $\hat{z}_{t|t-1}$ denotes the prediction of z_t ($\hat{z}_{t|t}$ has the analogous meaning) and $P_{t|t-1}$ is the error covariance matrix at time t, then given the state-space model for the time-series under consideration along with the initial estimate of the vector $\hat{z}_{0|0}$ and initial error covariance matrix $P_{0|0}$, the following steps are needed to generate the forecast of the process using Kalman filter (Kalman, 1963).

- *Prediction*

$$\hat{z}_{t|t-1} = \Phi z_{t-1|t-1} \tag{13}$$

$$P_{t|t-1} = \Phi P_{t-1|t-1}\phi^T + Q \tag{14}$$

- *Kalman gain computation*

$$L_t^* = P_{t|t-1}C^T\left[CP_{t|t-1}C^T + R\right]^{-1} \tag{15}$$

- *Update*

$$e_t = \{X_t - C\hat{z}_{t|t-1}\} \tag{16}$$

$$\hat{z}_{t|t} = \hat{z}_{t|t-1} + L_t^* e_t \tag{17}$$

$$P_{t|t} = \{I - L_t^* C\}P_{t|t-1} \tag{18}$$

Here I designates the identify matrix. The h-step forecast is obtained by

$$\hat{z}_{t+h|t} = \Phi^h \hat{z}_{t|t} \tag{19}$$

$$P_{t+h|t} = (\Phi^h)P_{t|t}(\Phi^h)^T + Q \tag{20}$$

Here h is kept fixed and t is varied in integer steps. In the offline analysis or in the situation, where one has access to the recorded data, the error covariance matrix $P_{t+h|t}$ plays an important role as the entries in the main diagonal represents the variance in the error, $e_t = z_t - \hat{z}_t$ during the prediction of states and hence the future outputs through the state-space model (Eqs. (1), (2)). Whereas, the quantity $\hat{z}_{t+h|t}$ is of more importance in the online estimation or in the situation, where one has to predict the behavior of the system in real time. Fig. 2 depicts the schematic of a block diagram of forecasting using Kalman filter.

For generating the forecast using Kalman filter algorithm, a second order state-space model of the form given in Eqs. (21), (22) is constructed.

$$\mathbf{z_t} = \begin{bmatrix}\phi_1 & \phi_2\\ 1 & 0\end{bmatrix}\mathbf{z_{t-1}} + \mathbf{w_t} \tag{21}$$

$$X_t = \begin{bmatrix}1 & 0\end{bmatrix}\mathbf{z_t} + \upsilon_t \tag{22}$$

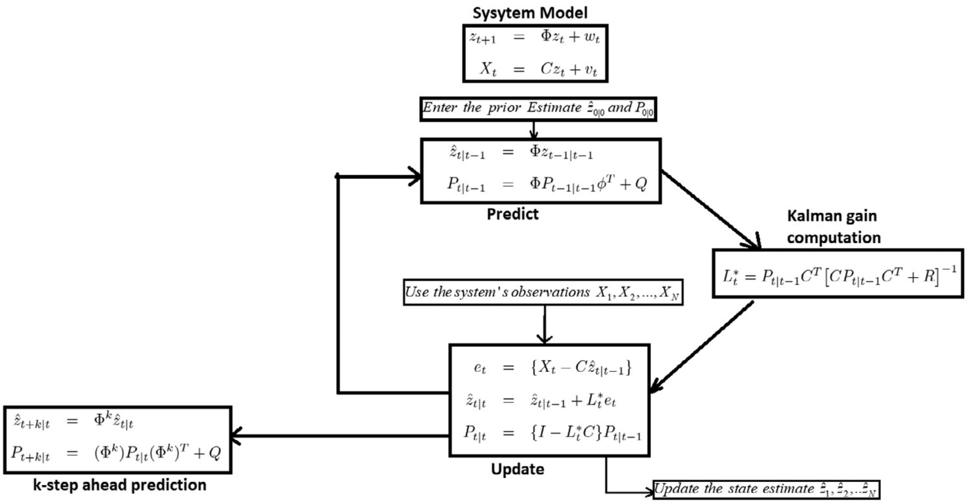
Sysytem Model
$z_{t+1} = \Phi z_t + w_t$
$X_t = Cz_t + v_t$
Enter the prior Estimate $\hat{z}_{0|0}$ and $P_{0|0}$
$\hat{z}_{t|t-1} = \Phi z_{t-1|t-1}$
$P_{t|t-1} = \Phi P_{t-1|t-1}\phi^T + Q$
Predict
Kalman gain computation
$L_t^* = P_{t|t-1}C^T\left[CP_{t|t-1}C^T + R\right]^{-1}$
Use the system's observations $X_1, X_2, ..., X_N$
$e_t = \{X_t - C\hat{z}_{t|t-1}\}$
$\hat{z}_{t|t} = \hat{z}_{t|t-1} + L_t^* e_t$
$P_{t|t} = \{I - L_t^* C\}P_{t|t-1}$
Update
$\hat{z}_{t+k|t} = \Phi^k \hat{z}_{t|t}$
$P_{t+k|t} = (\Phi^k)P_{t|t}(\Phi^k)^T + Q$
k-step ahead prediction
Update the state estimate $\hat{z}_1, \hat{z}_2, ...\hat{z}_N$

Fig. 2. Block diagram representing the estimation and forecasting by Kalman filter.

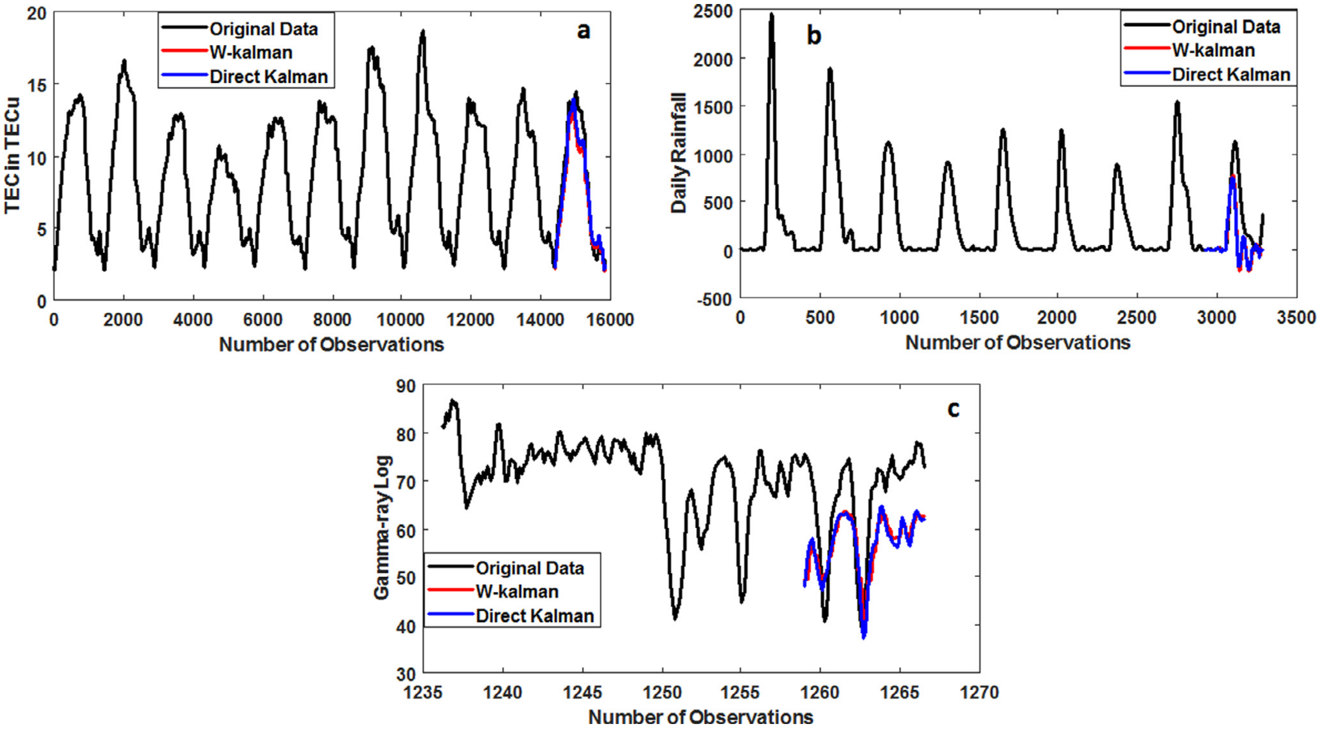

Fig. 3. Results of forecasting based on Kalman filter and W-Kalman for (a) TEC data for ALGO-2008 (b) rainfall data for Mumbai (c) geophysical well-log data.

Here $\mathbf{z_t} = \begin{bmatrix} X_t \\ X_{t-1} \end{bmatrix}$ is the vector with X_t being the value of the time-series at the time instant t. $\phi_1 \& \phi_2$ are the coefficient of the lag terms in the general AR-2 model given by

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \epsilon_t, \quad \epsilon_t \overset{iid}{\sim} \mathcal{N}(0, \sigma^2) \tag{23}$$

and $\mathbf{w_t} = \begin{bmatrix} \epsilon_t \\ 0 \end{bmatrix}$. Once the state-space model is formulated, we can then apply Eqs. (13)–(20) to generate the forecast of the vector $\hat{\boldsymbol{z}}_t$ and hence the future values of the process with $\boldsymbol{\Phi} = \begin{bmatrix} \phi_1 & \phi_2 \\ 1 & 0 \end{bmatrix}$ and $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$. In all the examples considered here, the initial value of the vector $(\hat{\mathbf{z}}_{0|0})$ is equal to the initial value of the time-series, the value of the initial error covariance matrix, $P_{0|0}$, is given by $P_{0|0} = \begin{bmatrix} 0 & 0 \\ 0 & \sigma^2 \end{bmatrix}$ and the covariance of input noise $\mathbf{w_t}$ is Q, defined as $Q = \begin{bmatrix} \sigma^2 & 0 \\ 0 & 0 \end{bmatrix}$, where σ^2 is the value of the innovation variance in the AR model.

In W-Kalman, a state-space model was estimated for the DWT coefficients of the training data. Using this state-space model, DWT coefficients of the test data were generated. The inverse wavelet transform algorithm was then used to reconstruct the time-series. The reconstructed time-series was then compared with the test data for the computation of RMSE and correlation coefficients. Fig. 3 and Tables 2, 3, 4 show the results of the forecast of different geophysical data sets obtained using Kalman filter and W-Kalman. It is evident from the results that W-Kalman produces better forecast than the general Kalman filter.

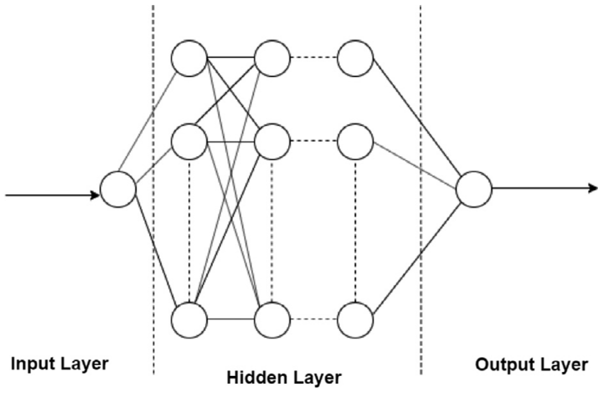

Fig. 4. A typical feedforward MLP neural network.

3.3. ANN and W-ANN algorithms for time-series forecasting

In ANN, the task of information processing is accomplished by neurons. Each neuron linearly combines several inputs it receives from other neurons in the network and processes the output by a non-linear transfer function (Sarle, 1994). ANNs are fast and are capable of learning important features from the training data set and generalizing the results for the test data. Among different architectures of ANN that have been proposed since their inception (Hopfield, 1982), the MLP architecture is the most suited one for forecasting (Park et al., 1991; Hill et al., 1994, 1996). MLP is used in a variety of contexts especially in forecasting problems because of their inherent capability of learning arbitrary input–output mapping that exist in the training data. Fig. 4 shows a schematic sketch of an MLP architecture that is generally used for forecasting. A typical MLP is composed of many layers. The first layer is called the input layer, which receives the external information. The last layer is the output layer, where the solution to the problem is obtained. Between the input and output layers, there may be one or more hidden layers. Different nodes in the adjacent layers are fully connected. For time-series forecasting the inputs to the network consist of the past values of the time-series $X_t, X_{t-1}, X_{t-2} \dots, X_{t-p+1}$ and the output is the future value X_{t+1}. So, the ANN performs the following input/output mapping

$$X_{t+1} = f(X_t, X_{t-1}, \dots, X_{t-p+1})$$

Thus, the ANN implements a non-linear autoregressive model for the time-series under consideration. The training procedure of an MLP, which is called supervised learning is as follows:

First, the examples of a training set are entered into the input nodes. The activation values of the input nodes are linearly combined and then a non-linear transformation takes place at each node in the first hidden layer. After transformation, these values become the input to the nodes in the next adjacent hidden layer. This process is repeated until the output activation values are found. The training algorithm is used to find the appropriate connecting weights such that a certain measure of error like sum of squared error (SSE) or mean squared error (MSE) is minimized. Hence, the training algorithm is used to solve an unconstrained optimization problem.

Some of the important functions used in the transformations carried out by the neurons are:

- The sigmoid(logistic) function: $f(x) = \frac{1}{1+e^{-x}}$
- The hyperbolic tangential function: $f(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
- the sine or cosine function: $f(x) = \sin x$ or $f(x) = \cos x$
- linear function: $f(x) = x$

In the present study, the procedure for training and testing of the ANN is done as follows. Given the time series data $X = [X_1, X_2, \dots, X_N]$, we divide it into k non-overlapping segments of length L_w, where $k = \frac{N}{L_w}$. Next, we derive four different vectors from the given time-series data to train the network, namely,

$$\begin{aligned}
X_{train} &= [X_1, X_2, \dots, X_{\frac{N(k-2)}{k}}]^T \\
y_{train} &= [X_{\frac{N}{k}+1}, X_{\frac{N}{k}+2}, \dots, X_{\frac{N(k-1)}{k}+1}]^T \\
X_{test} &= [X_{\frac{N(k-2)}{k}+1}, X_{\frac{N(k-2)}{k}+2}, \dots, X_{\frac{N(k-1)}{k}}]^T \\
y_{test} &= [X_{\frac{(k-1)N}{k}+1}, X_{\frac{(k-1)N}{k}+2}, \dots, X_N]^T
\end{aligned}$$

Then, the i^{th} input–output data pair during training will be of the form,

$$(X_{train})_i = X_{\frac{(i-1)N}{k}+1} \Leftrightarrow (y_{train})_i = X_{\frac{iN}{k}+1} \quad \text{for, } \; i = 1, \dots, k-1.$$

Next, the MLP architecture is constructed, where the input and output layer has 1 neuron. The architecture contains M hidden layers, each containing $h_1, h_2, \dots, h_M$ number of neurons. The tanh activation function is used for all hidden layers except for the output layer, which has linear activation function. The output of such a network for an input X_i is given by,

$$\hat{y}_i = F(W|X_i) \tag{24}$$

$$\hat{y}_i = W_{M0}(\tanh(..W_{23}(\tanh(W_{12}(\tanh(W_{I1}X_i + b_1)) + b_2)) + b_3)..) + b_0 \tag{25}$$

where, $W_{I1}, b_1 \in \mathbf{R}^{h_1}$, $W_{12} \in \mathbf{R}^{h_2 \times h_1}, b_2 \in \mathbf{R}^{h_2}$, $W_{23} \in \mathbf{R}^{h_3 \times h_2}, b_3 \in \mathbf{R}^{h_3}$, $\dots$, $W_{M0} \in \mathbf{R}^{1 \times h_M}, b_0 \in \mathbf{R}$. W represents the set containing all the free parameters (weights and biases). Once the network is trained using X_{train} and y_{train}, then the entries of X_{test} vector are used to test the network. The results are compared to the entries of the vector y_{test} which is the ground truth. During training, the following cost function is minimized

$$J(W|X_{train}, y_{train}) = \frac{1}{(k-2)\frac{N}{k}} \sum_{i=1}^{(k-2)\frac{N}{k}} (y_{train} - \hat{y}_{train})^2$$

where, $\hat{y}_{train} = F(W|X_{train})$, The parameters W are adjusted till the above cost function is minimized. Therefore, the optimal parameter W^* is defined as

$$W^* = \arg\min_W J(W|X_{train}, y_{train})$$

The above parameters are optimized using Levenberg–Marquardt optimization algorithm. During testing, the entries of the X_{test} vector are used for generating the output $\hat{y}_{test}$, which is compared with the vector y_{test} to calculate the RMSE and correlation coefficients.

For generating the forecast of TEC, rainfall and geophysical well-log data using ANN, a fully connected multilayer feed forward ANN was used. ANN for TEC data has four hidden layers with 5, 12, 15 and 7 neurons along with an input and output layer. During the training process, first 1440 (1-day cycle) values of TEC time-series were used as input and the second 1440 (2nd day cycle) values were used as target values. The connection weights were adjusted till the minimum $\mathcal{L}^2$ norm of error was achieved. Then the values of the 2nd cycle were used as input and the values of the 3rd cycle as target. This process was repeated for the first nine cycles as input. Once the training was completed, then the neural network was simulated using the tenth cycle as input to produce the forecast for the eleventh cycle. Similar process was followed for predicting the values of rainfall data and well-log data.

In W-ANN the training and testing is analogous to normal ANN, wherein, the DWT coefficients of the training data are used to predict the DWT coefficients of the testing data. Inverse wavelet transform is then used to generate the values of the time domain samples.

Fig. 5 shows the results of the forecast of different geophysical data sets obtained using ANN and W-ANN. It is evident that W-ANN produces better forecast than the simple ANN. Also, from Figs. 1, 3, 5 and Tables 2, 3, 4, it is evident that the wavelet versions of different forecasting algorithms produce better forecasts than their regular counterparts. More details of these results are discussed in the next section.

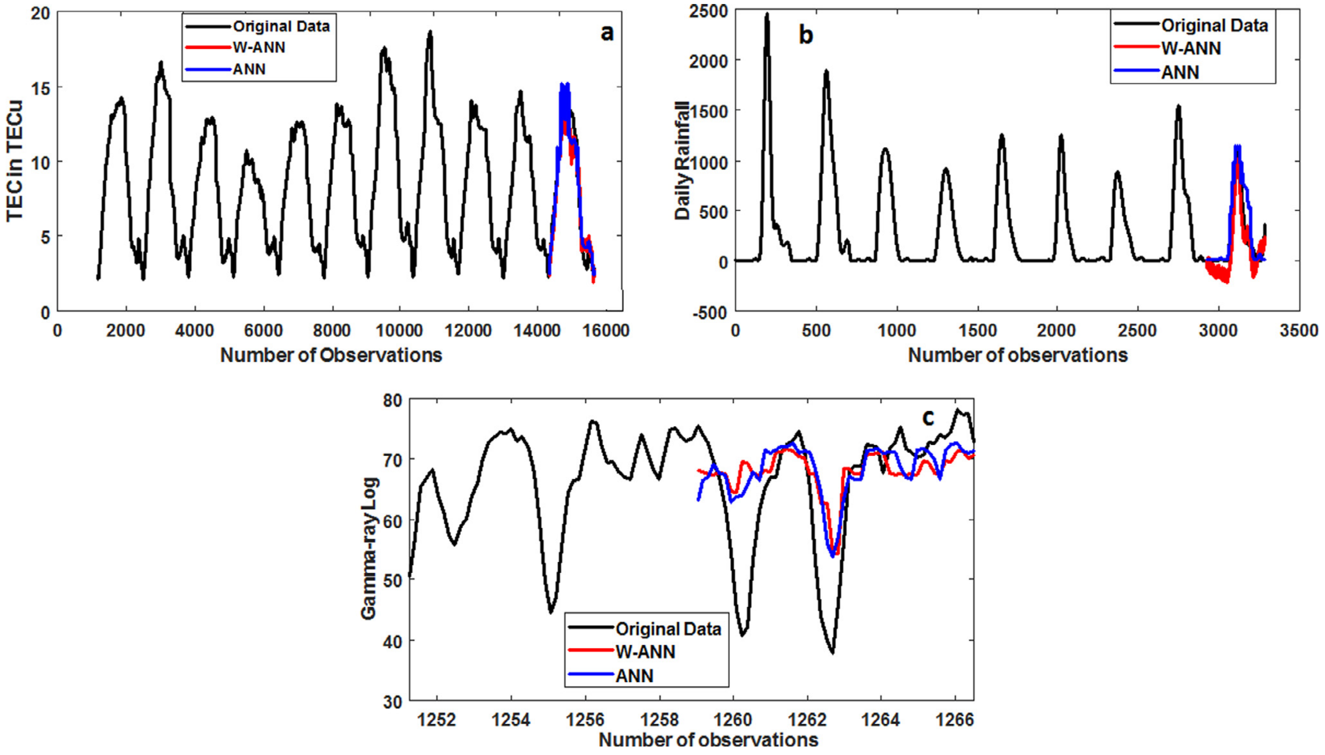

Fig. 5. Results of forecasting based on ANN and W-ANN approach for (a) TEC data for ALGO-2008 (b) rainfall data for Mumbai (c) geophysical well-log data.

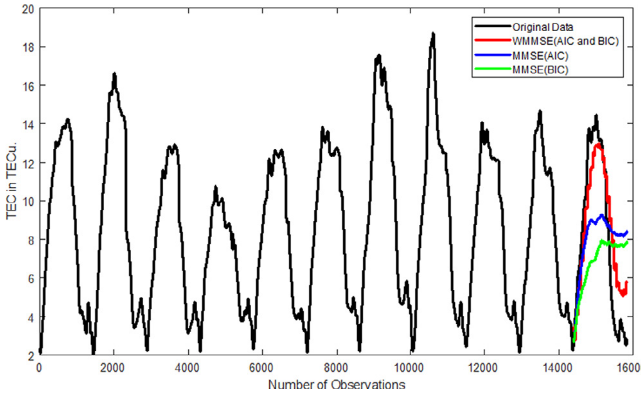

Fig. 6. Results of forecasting of TEC data corresponding to the station ALGO in the year 2008.

4. Results and discussion

ANN being a non-linear data driven approach, is better suited to capture the non-linearity present in the system generating the data, since it performs the non-linear auto regression of the past values to predict the future values of the process. One way to enhance the performance of the non-linear regression is to use the wavelet coefficients of the data as the input feature, rather than discrete values of the raw time-series data. This is because, the wavelet coefficients efficiently capture the discontinuities and other high frequency features present in the data. Therefore, if the feed forward ANN is trained using the wavelet coefficients of the training data to predict the wavelet coefficients of the test data, then the prediction might be able to possess all the features present in the original data.

Figs. 1a, 3a and 5a show the results of different forecasting techniques on the TEC data for ALGO 2008. Here different forecasting algorithms were used to forecast 1440 (1 day) observations (test data) for TEC using past 14,400 (10-days) observations (training data). Figs. 1b, 3b and 5b show the results of different forecasting techniques on the daily rainfall data of Mumbai. Here different forecasting algorithms were used to forecast 365 (1 year) observations (test data) for the amount of rainfall values using past 2920 (8-years) observations (training data). Similarly, Figs. 1c, 3c and 5c show the results of different forecasting techniques applied to the gamma-ray log data of an offshore oil basin. Here 50 values corresponding to data up to a depth of 16.3 meters (test data), were predicted using the past 1950 values corresponding to a depth of 650 meters (training data).

Figs. 1b, 3b, 5b show the comparison of forecast generated using Kalman filter and W-Kalman. It is evident from the above figures and Tables 2–4 that, W-Kalman performs better than its regular counterpart. The better performance of W-MMSE and W-Kalman over their respective regular counterparts can be attributed to the fact that these models are trained using the DWT coefficients of training data set. If the DWT is performed using a properly chosen wavelet basis, then the DWT coefficients are wide-sense stationary processes. Since ARIMA processes are inherently wide-sense stationary, one would expect ARIMA models to provide a better fit for the DWT coefficients. Since Kalman filters are based on the state-space models derived from the ARIMA models, the W-Kalman produces better forecast than their regular counterparts.

Fig. 5 shows the comparison of forecasting performances of ANN and W-ANN. While the values of the time-series were directly used for training and testing the performance of ANN, the DWT coefficients of part of the time-series were used to predict the DWT coefficients of the consecutive part of the data with the same length in case of W-ANN. The inverse wavelet transform was then used to reconstruct the predicted time-series. Haar wavelet decomposition coefficients at level-6 were used in W-ANN to predict TEC and Rainfall data and level-1 decomposition coefficients were used to predict the well-log data. In ANN, although a large number of free parameters may increase the accuracy of the prediction, it may sometimes lead to over-fitting. In the present study, we have selected such an architecture, which has given the best results with least error on the test-data. Therefore, the problem of overfitting of the data does not arise in our present analysis of forecasting.

For a time-series data of length N and a given forecast horizon h, the performance measure of different forecasting techniques was compared using the metrics, RMS error (RMSE) and correlation coefficient (r).

$$RMSE = \sqrt{\frac{1}{h}\sum_{i=1}^{h}(\hat{X}_{N+i} - X_{N+i})^2} \tag{26}$$

$$r = \frac{\sum_{i=N+1}^{N+h}(\hat{X}_i - \mu_{\hat{X}})(X_i - \mu_X)}{\sqrt{\sum_{i=N+1}^{N+h}(\hat{X}_i - \mu_{\hat{X}})^2}\sqrt{\sum_{i=N+1}^{N+h}(X_i - \mu_X)^2}} \tag{27}$$

where $\mu_{\hat{X}} = \frac{1}{h}\sum_{i=1}^{h}\hat{X}_{N+i}$ and $\mu_X = \frac{1}{h}\sum_{i=1}^{h}X_{N+i}$ are the sample means for the predicted and the original data over the prediction horizon. Tables 2–4 show the values of the RMSE and the correlation coefficient for the TEC, rainfall and well-log data respectively.

5. Conclusions

Most of the geophysical systems are highly non-linear and chaotic in nature. As a result, the data generated by these processes display a complicated correlation structure between successive observations. From the results it is evident that, forecasting using ARIMA(p,d,q) models and the Kalman filter exhibits poor performance, when the direct time-series values are used to forecast the data. This is because of the underlying assumption that the data under consideration has a linear additive model and a linear state-space model. Also, the nature of the innovation sequence in the case of ARIMA process and the input noise and the observation noise sequences in the Kalman filter model have simple statistics. They represent wide-sense stationary Gaussian process. These assumptions sometimes lead to over simplification of the complex dynamics that a geophysical system exhibits. However, when ARIMA (p,d,q) models with W-MMSE and W-Kalman were used to generate the forecast of the wavelet coefficients, their forecasting performance improved significantly, as can be seen from the values of the RMSE and correlation coefficients. This can be attributed to the fact that wavelet transform decorrelates the data and removes the non-stationary features of the data in the wavelet domain. Also, since the Kalman filter is based on the state-space models derived from ARIMA models, the performance of W-MMSE and W-Kalman improves over their regular counterparts. If one wants to predict h points ahead in the time-series, then this translates to predicting $\lceil h/2 \rceil$ points at level-1, $\lceil h/4 \rceil$ points at level-2 and so on. Generally, as the forecast horizon increases, the variance of the prediction error also increases. However, in case of wavelet-based methods, the prediction accuracy improves because of the inherent short horizon forecasting. ANN on the other hand being a self-driven data adaptive methodology, contains hidden layers with a lot of connecting weights, which along with the incorporation of the non-linear transfer function enables ANN to effectively capture the non-linearity in the geophysical system generating the data. These features are enhanced further using W-ANN, where the wavelet coefficients are considered as input features instead of the raw time-series data for the purpose of training a feed-forward ANN. Finally, we believe that W-ANN gives the best prediction of nonlinear data, when compared with all other algorithms discussed in this paper.

6. Computer code availability

All the necessary codes for implementing different forecasting algorithms were developed in MATLAB 2018. The econometrics toolbox of MATLAB was used for developing an ARIMA(p,d,q) model for a given time-series data. AIC and BIC were also calculated using the econometrics toolbox. Kalman filter algorithm was implemented in MATLAB using the elementary linear algebra subroutines. The deep learning toolbox of MATLAB 2018, was used to construct and train the feed forward neural network. Wavelet toolbox of MATLAB 2018 was used to implement the discrete wavelet transform. Different functions from econometrics toolbox and wavelet toolbox were jointly used to implement wavelet-based version of the different forecasting algorithms. All the subroutines developed in MATLAB are available for download at https://github.com/shivamjaipurwale/PAPER2.

Declaration of competing interest

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.

Acknowledgments

The authors thank the handling editor and four anonymous referees for their meticulous reviews, which have improved the quality of the paper. This work forms a part of the Ph.D thesis of SB. SB thanks IIT Bombay for providing him with the necessary fellowship for pursuing his Ph.D.

References

Akaike, H., 1969. Fitting autoregressive models for prediction. Ann. Inst. Statist. Math. 21 (1), 243–247.

Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M., 2015. Time Series Analysis: Forecasting and Control. John Wiley & Sons.

Chandrasekhar, E., Prabhudesai, S.S., Seemala, G.K., Shenvi, N., 2016. Multifractal detrended fluctuation analysis of ionospheric total electron content data during solar minimum and maximum. J. Atmos. Sol.-Terr. Phys. 149, 31–39.

Chandrasekhar, E., Rao, V.E., 2012. Wavelet analysis of geophysical well-log data of bombay offshore basin, India. Math. Geosci. 44 (8), 901–928.

Chen, Y., Yang, B., Dong, J., 2006. Time-series prediction using a local linear wavelet neural network. Neurocomputing 69 (4–6), 449–465.

Coulibaly, P., Baldwin, C.K., 2005. Nonstationary hydrological time series forecasting using nonlinear dynamic methods. J. Hydrol. 307 (1–4), 164–174.

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Math. Control Signals Syst. 2 (4), 303–314.

Daubechies, I., 1992. Ten Lectures on Wavelets, Vol. 61. Siam.

Doucoure, B., Agbossou, K., Cardenas, A., 2016. Time series prediction using artificial wavelet neural network and multi-resolution analysis: Application to wind speed data. Renew. Energy 92 (3), 202–211.

Durbin, J., Koopman, S.J., 2002. A simple and efficient simulation smoother for state space time series analysis. Biometrika 89 (3), 603–616.

Durbin, J., Koopman, S.J., 2012. Time Series Analysis by State Space Methods. Oxford university press.

Funahashi, K.-I., 1989. On the approximate realization of continuous mappings by neural networks. Neural Netw. 2 (3), 183–192.

Gers, F.A., Eck, D., Schmidhuber, J., 2002. Applying LSTM to time series predictable through time-window approaches. In: Neural Nets WIRN Vietri-01. Springer, pp. 193–200.

Hagan, M.T., Menhaj, M.B., 1994. Training feedforward networks with the Marquardt algorithm. IEEE Trans. Neural Netw. 5 (6), 989–993.

Hamilton, J.D., 1994. Time Series Analysis, Vol. 2. Princeton university press, Princeton, NJ.

Hecht-Nielsen, R., 1992. Theory of the backpropagation neural network. In: Neural Networks for Perception. Elsevier, pp. 65–93.

Hill, T., Marquez, L., O'Connor, M., Remus, W., 1994. Artificial neural network models for forecasting and decision making. Int. J. Forecast. 10 (1), 5–15.

Hill, T., O'Connor, M., Remus, W., 1996. Neural network models for time series forecasts. Manag. Sci. 42 (7), 1082–1092.

Holloway Jr., J.L., 1958. Smoothing and filtering of time series and space fields. In: Advances in Geophysics, Vol. 4. Elsevier, pp. 351–389.

Hopfield, J.J., 1982. Neural networks and physical systems with emergent collective computational abilities. Proc. Natl. Acad. Sci. 79 (8), 2554–2558.

Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Zheng, Q., Yen, N.-C., Tung, C.C., Liu, H.H., 1998. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. In: Proceedings of the Royal Society of London a: Mathematical, Physical and Engineering Sciences, Vol. 454. (1971), The Royal Society, pp. 903–995.

Kailath, T., 1980. Linear Systems, Vol. 156. Prentice-Hall, Englewood Cliffs, NJ.

Kalman, R.E., 1960. A new approach to linear filtering and prediction problems. J. Basic Eng. 82 (1), 35–45.

Kalman, R.E., 1963. Mathematical description of linear dynamical systems. J. Soc. Ind. Appl. Math. Ser. A: Control 1 (2), 152–192.

Klobuchar, J., Parkinson, B., Spilker, J., 1996. Ionospheric Effects in Global Positioning System: Theory and Applications, Vol. 1. American Institute of Aeronautics and Astronautics, Washington DC.

Längkvist, M., Karlsson, L., Loutfi, A., 2014. A review of unsupervised feature learning and deep learning for time-series modeling. Pattern Recognit. Lett. 42, 11–24.

Luo, X., Bhakta, T., Jakobsen, M., Nævdal, G., 2016. An ensemble 4d seismic history matching framework with sparse representation based on wavelet multiresolution analysis. SPE J. 22 (03), 985–1010.

Mallat, S.G., 1989. Multiresolution approximations and wavelet orthonormal bases of $\mathcal{L}^2(\mathbf{R})$. Trans. Amer. Math. Soc. 315 (1), 69–87.

Nason, G.P., Von Sachs, R., 1999. Wavelets in time-series analysis. Philos. Trans. R. Soc. Lond. A 357 (1760), 2511–2526.

Park, D.C., El-Sharkawi, M., Marks, R., Atlas, L., Damborg, M., 1991. Electric load forecasting using an artificial neural network. IEEE Trans. Power Syst. 6 (2), 442–449.

Partal, T., Cigizoglu, H.K., 2009. Prediction of daily precipitation using wavelet—neural networks. Hydrol. Sci. J. 54 (2), 234–246.

Percival, D.B., Walden, A.T., 2006. Wavelet Methods for Time Series Analysis, Vol. 4. Cambridge university press.

Posada, D., Buckley, T.R., 2004. Model selection and model averaging in phylogenetics: advantages of akaike information criterion and Bayesian approaches over likelihood ratio tests. Syst. Biol. 53 (5), 793–808.

Qiu, X., Zhang, L., Ren, Y., Suganthan, P.N., Amaratunga, G., 2014. Ensemble deep learning for regression and time series forecasting. In: 2014 IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL). IEEE, pp. 1–6.

Sarle, W.S., 1994. Neural Networks and Statistical Models. Citeseer.

Seemala, G., Valladares, C., 2011. Statistics of total electron content depletions observed over the south american continent for the year 2008. Radio Sci. 46 (5).

Sharma, M., Vanmali, A.V., Gadre, V.M., 2013. Construction of wavelets: Principles and practices. In: Chandrasekhar, E., Dimri, V., Gadre, V. (Eds.), Wavelets and Fractals in Earth System Sciences. Taylor & Francis, CRC Press, pp. 29–92.

Specht, D.F., 1991. A general regression neural network. IEEE Trans. Neural Netw. 2 (6), 568–576.

Stolojescu, C., Railean, I., Lenca, S.M.P., Isar, A., 2010. A wavelet based prediction method for time series. In: Proceedings of the 2010 International Conference Stochastic Modeling Techniques and Data Analysis, Chania, Greece, pp. 8–11.

Wold, H., 1938. A Study in the Analysis of Stationary Time Series (Ph.D. thesis). Almqvist & Wiksell.

Yao, S., Hu, S., Zhao, Y., Zhang, A., Abdelzaher, T., 2017. Deepsense: A unified deep learning framework for time-series mobile sensing data processing. In: Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, pp. 351–360.

Yousefi, S., Weinreich, I., Reinarz, D., 2005. Wavelet-based prediction of oil prices. Chaos Solitons Fractals 25 (2), 265–275.

Zhang, G., Patuwo, B.E., Hu, M.Y., 1998. Forecasting with artificial neural networks:: The state of the art. Int. J. Forecast. 14 (1), 35–62.