Neurocomputing 361 (2019) 151–163

Contents lists available at ScienceDirect

Neurocomputing

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

Improving forecasting accuracy of time series data using a new ARIMA-ANN hybrid method and empirical mode decomposition

Ümit Çavuş Büyükşahin[a,*], Şeyda Ertekin[a,b]

[a] Department of Computer Engineering, Middle East Technical University (METU), Üniversiteler Mh., No:1, Ankara 06800, Turkey
[b] MIT Sloan School of Management, Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 02139, USA

ARTICLE INFO

Article history:
Received 17 July 2018
Revised 1 February 2019
Accepted 3 May 2019
Available online 16 July 2019

Communicated by Prof. A. Prieto

Keywords:
Time series forecasting
ANN
ARIMA
EMD
Hybrid methods

ABSTRACT

Many applications in different domains produce large amount of time series data. Making accurate forecasting is critical for many decision makers. Various time series forecasting methods exist that use linear and nonlinear models separately or combination of both. Studies show that combining of linear and nonlinear models can be effective to improve forecasting performance. However, some assumptions that those existing methods make, might restrict their performance in certain situations. We provide a new Autoregressive Integrated Moving Average (ARIMA)-Artificial Neural Network (ANN) hybrid method that work in a more general framework. Experimental results show that strategies for decomposing the original data and for combining linear and nonlinear models throughout the hybridization process are key factors in the forecasting performance of the methods. By using these findings, the proposed hybrid method is combined with Empirical Mode Decomposition (EMD) technique which generates more predictable components. We show that our hybrid method with EMD can be an effective way to improve forecasting accuracy obtained by traditional hybrid methods and also any of the individual methods that we used separately.

1. Introduction and related work

Time series include data points listed in time order. It is generally a sequence of discrete-time data which consists of points equally spaced in time. In time series forecasting, we try to predict the future points by analyzing observed points in the series. It has been widely used in various applications of science, engineering, and business fields. However, time series data might show different characteristics and show increasing or decreasing trends. Some time series data have seasonal trends in which variations are specific to a particular time range, e.g coat and boot sales increase in winter season while decrease in summer season. On the other hand, some time series data are not seasonal, such as stock market data. Moreover, time series data might show different level of volatility. For example, USD/EUR (United States dollar/Euro) exchange rate shows high volatility, on the other hand, growth of an animal, plant, or human being show a linear change. Over past several decades, a considerable effort has been devoted to develop and improve time series forecasting models [1]. In the literature, various forecasting methods have been proposed which use linear and nonlinear models separately or combination of both. In this paper, we propose a hybrid algorithm of linear and nonlinear methods where we choose Autoregressive Integrated Moving Average (ARIMA) as a linear method and Artificial Neural Networks (ANNs) as a nonlinear method. Furthermore, the performance of the proposed method is enhanced by using a multi-scale decomposition technique such as Empirical Mode Decomposition (EMD).

ARIMA is widely used linear time series forecasting method that is used in numerous applications including finance [2], engineering [3], social sciences [4], and agriculture [5]. ARIMA models are integration of Autoregressive models (AR) and Moving Average models (MA). ARIMA models give good accuracy in forecasting relatively stationary time series data. However it makes a strong assumption that the future data values are linearly dependent on the current and past data values. Therefore, many real world time series data presents complex nonlinear patterns which might not be modeled by ARIMA effectively.

For the nonlinear time series modeling, Artificial Neural Networks (ANNs) are one of the most widely used algorithms [6] in many fields, such as finance [7], energy [8], hydrology [9], and network communications [10]. ANNs have several advantages over ARIMA and other forecasting models. Firstly, ANNs are capable of fitting a complex nonlinear function. This ability helps ANNs to approximate any continuous measurable function with arbitrarily

* Corresponding author.
E-mail addresses: umit.buyuksahin@ceng.metu.edu.tr (Ü.Ç. Büyükşahin), seyda@ceng.metu.edu.tr (Ş. Ertekin).

https://doi.org/10.1016/j.neucom.2019.05.099
0925-2312/

desired accuracy [11,12]. Moreover, ANNs are adaptively data-driven in nature which means ANN models can be adaptively formed based on the features of time series data [13].

In the literature, there are studies which show the success of linear and nonlinear methods over each other. For example, [14–16] report that statistical and linear models give better results than ANNs. On the other hand, [17–19] report that ANN performs better than linear models when data exhibit high volatility and multicollinearity.

In short, each type of model might not perform well in all circumstances. In order to overcome this limitation, various hybrid techniques have been proposed which aim to take advantage of the unique strength of each different type of models.

The common practice in hybrid techniques is to decompose time series data into its linear and nonlinear forms, then use appropriate type of models on them separately. A hybrid ANN-ARIMA model proposed by Zhang [13] achieves more accurate forecast results in time series data as compared to using individual models, in applications such as electricity price forecasting [20] and water quality prediction [21]. Another successful hybrid ARIMA-ANN technique is presented by Khashei and Bijari [22] which defines functional relationship between components. Moreover, Babu and Reddy [23] offers a solution to volatility problem in time series data by smoothing out dataset with moving average filter.

Each hybrid method in the literature bring different perspectives to time series forecasting problem. However strong assumptions that these methods make might degenerate their performances if the contrary situations occur. The strength of the hybrid methods comes from treating the linear and nonlinear components of the time series in different ways. Therefore, proper decomposition of series is critical. However [13] and [22] do not decompose time series data into linear and nonlinear components. Rather, they assume that linear component of the data is the output of the ARIMA model. Moreover, [13] and [23] assume that the output of their hybrid methods is linear combination of the components. However, different datasets might suggest different type of relationships between the output and the components.

In this study, we propose a novel hybrid method for time series forecasting which aims to overcome the limitations of the traditional hybrid methods by eliminating the need to make strong assumptions. In this method, nature of nonlinearity is first characterized by the help of moving-average (MA) filter, then ARIMA is applied to the linear component. In the final step, ANN is used to combine the output of ARIMA, the nonlinear component, and the original data. By this means, we do not make any assumptions in the extraction of components and in the modeling of components. Three benchmark datasets, the Wolf's sunspot data, the Canadian lynx data, and the British pound/US dollar exchange rate data and an additional public dataset, Turkey Intraday Electricity Market Price data are used in order to show the effectiveness of the proposed method in time series forecasting.

In addition, we propose an improvement to all hybrid methods mentioned in this paper, including ours, by adding Empirical Mode Decomposition (EMD) technique [24] to the models. When accuracy results of the hybrid methods with different types of datasets are compared, we observe that accuracy performance gets better with the increasing level of linearity in time series. Then, time series data can be considered as a merge of sub-series which each of them demonstrates more linearity. For this purpose, a well-known multiscale decomposition technique, EMD is used in the proposed hybrid method. The components achieved by EMD are relatively stationary and have simpler frequency range which make them strongly correlated in themselves. Thus, more accurate predictions can be obtained through the models [25,26].

The rest of the paper is organized as follows. In the next section, we present existing different time series forecasting methods. In Section 3 we present our proposed model. In Section 4, we show the evaluation results of our model and present comparison results with the other methods. In Section 5, we provide discussions and present the positive effect of using EMD technique with the hybrid methods and our new method. We show that our method consistently outperforms other linear, nonlinear and even hybrid methods.

2. Time series forecasting methods

In order to give overall review before we present our method, we want to give a brief information about other time series forecasting methods, such as ARIMA, ANN, and the well-known hybrid methods. We chose three successive hybrid methods where each one has taken previous one(s) as reference study. Since benchmark datasets are used in the evaluation of these hybrid methods, we were able to repeat their studies for our comparisons.

2.1. Autoregressive integrated moving average method

ARIMA is a linear method which means future value of a variable to be forecasted is assumed to be linear function of the past observations. As a consequence, time series data that is fed to ARIMA is expected to be linear and stationary.

ARIMA performs stationarity check on given time series data to check whether mean and autocorrelation patterns are constant over time. If stationary property is not satisfied, ARIMA applies differencing method d times until non-stationary property is disposed. As a consequence, the integration order of ARIMA model is set to be d. Thereafter, an autoregressive moving average(ARMA) is applied on the resultant data as follows:

Let the actual data value is y_t and random error ϵ_t at any given time t. This actual value y_t is considered as a linear function of the past p observation values, say $y_{t-1}, y_{t-2}, \ldots, y_{t-p}$ and q random erros, say $\epsilon_t, \epsilon_{t-1}, \ldots, \epsilon_{t-q}$. The corresponding ARMA equation is given in the following equation:

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \cdots + \alpha_p y_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_q \epsilon_{t-q} \quad (1)$$

In Eq. (1), the coefficients from α_1 to α_p are Autoregression coefficients, θ_1 to θ_q are Moving Average coefficients. Note that random errors ϵ_t are identically distributed with a mean of zero and a constant variance.

Similar to the d parameter, p and q coefficients are referred to as the orders of the model. When q equals to zero, the model is reduced to AR model of order p (AR(p)). If p is equal to zero, the model becomes MA model of order q (MA(q)).

2.2. Artificial neural networks method

ANN provides flexible computation framework for nonlinear modeling in wide range of applications. Due to its flexible architecture, number of layers and the neurons at each layer can be easily varied. In addition, ANN does not require any prior assumption, such as input data stationarity. ANN network configuration is largely determined by the characteristics of the data.

The architecture of the most widely used ANN models in time series forecasting contains three-layers. The neurons of layers are acyclically linked. In order to model time series data using such a three-layer network, nonlinear function f of y_t sequence from y_{t-1} to y_{t-N} is constructed as shown in the following equation:

$$y_t = w_0 + \sum_{j=1}^{H} w_j f\left(w_{0j} + \sum_{i=1}^{N} w_{ij} y_{t-i} \right) + e_t \quad (2)$$

where, at any given time t, w_{ij} and w_j are model weights and H and N are the number of hidden and input nodes, respectively. In this equation, e_t corresponds to a noise or error term. The transfer function of the hidden layers f in ANN architecture can be functions such as sigmoid, ReLu, tanh, etc.

2.3. Zhang's hybrid method

For time series forecasting, Zhang proposed a hybrid ARIMA-ANN model [13]. According to this model, it is assumed that time series data is a sum of linear and nonlinear components, given in the form of:

$$y_t = L_t + N_t \tag{3}$$

where L_t denotes the linear and N_t denotes the nonlinear component. Firstly, ARIMA is used with the given time series data and linear forecasts are obtained. Residuals from linear component is assumed to contain only nonlinear relationship. This method uses ARIMA to make forecast from the linear component and ANN from the nonlinear component. Then, these models are combined to improve overall forecasting performance. This method gives better foreceasting accuracy than using ARIMA and ANN methods individually, as seen in the experimental results in three well-known real data sets - the Wolf's sunspot data, the Canadian lynx data, and the British pound/US dollar exchange rate data.

2.4. Khashei and Bijari's hybrid method

For time series forecasting, Khashei and Bijari proposed another hybrid ARIMA-ANN [22]. Similar to Zhang's model, this model also assumes that any time series data is composed of linear and nonlinear components. Likewise, ARIMA is used to extract linear component and make forecast on it and residuals, which are nonlinear components, are fed into ANN along with the original data, and linear forecast of ARIMA output. The difference from the Zhang's model is to avoid the assumption that the relationship between linear and nonlinear components is additive. Rather, this method builds functional relationship between the components as shown in the following equation:

$$y_t = f(L_t, N_t) \tag{4}$$

where L_t is the linear and N_t is the nonlinear component.

In addition, one may not guarantee that the residuals of the linear component may comprise valid nonlinear patterns. Khashei and Bijari suggest that residuals should not put into ANN as an input alone.

2.5. Babu and Reddy's hybrid method

The hybrid ARIMA-ANN method proposed by Babu and Reddy [23] integrates moving average filter into hybrid ANN-ARIMA model. Like other methods, this model also assumes that any time series data is composed of linear and nonlinear components. However, this study emphasizes that neither Zhang nor Khashei and Bijari's methods decompose original time series data into its linear and nonlinear components; instead, they use a linear ARIMA model to extract the linear component and the error sequences is assumed to be nonlinear component. On the other hand, this study separates the linear and nonlinear components, then feed them into appropriate methods.

This method tries to fix Moving Average (MA) filter length until *kurtosis* value of the data becomes approximately 3. Kurtosis is a shape of a probability distribution which measure thickness or heaviness of the tails of a distribution. The kurtosis value is 3 if the data has normal distribution. Shortly, the method aims to find out normal distributed component, which shows low volatility, in time series data by using kurtosis value. When the low-volatile component is separated from the original data, high-volatile component, which is assumed to be nonlinear, is achieved. In the final step, like in Zhang's method, the decomposed components are fed into ARIMA and ANN accordingly and forecast results are summed up to achieve final forecast.

3. Proposed method

Many decision processes need high forecasting accuracies in time series applications. Although there are numerous available time series models, none of them consistently gives the best results in various situation. There are two main challenges for making an accurate forecast. The first challenge is that underlying data generating process of time series cannot easily identified [27]. The second one is that non-hybrid individual models are generally insufficient to determine all the characteristics of the time series [13]. Many researches in time series forecasting literature show that hybrid models improve the forecasting performances [28]. By taking the advantage of each individual method in a combined model, error risk of using an inappropriate method is reduced and more accurate results are obtained.

Each hybrid method mentioned in this paper bring different perspectives into time series forecasting problem. However, strong assumptions these methods make might degenerate their performances in certain circumstances. In this study, we propose to novel hybrid method for time series forecasting which aims to overcome the limitations of traditional hybrid methods by eliminating strong assumptions. The architecture of the proposed hybrid method is shown in Fig. 1.

The algorithm starts with data decomposition. In this method time series data y_t is considered as a function of linear L_t and nonlinear N_t components in the same way as given in Eq. (4).

These two components are separated from the original data by using moving average (MA) filter with the length of m, as given in Eq. (5). While the linear component l_t has low volatility, the residual r_t, which is the difference between the original data and the decomposed linear data in Eq. (6), shows high fluctuation.

$$l_t = \frac{1}{m} \sum_{i=t-m+1}^{t} y_i \tag{5}$$

$$r_t = y_t - l_t \tag{6}$$

In order for a proper decomposition, the length of the MA filter m has to be adjusted. Augmented Dickey Fuller (ADF) test which is unit root test can be performed to determine whether a given data series is stationary or not. The existence of a unit root on a given dataset indicates that there is an unpredictable systematic pattern.

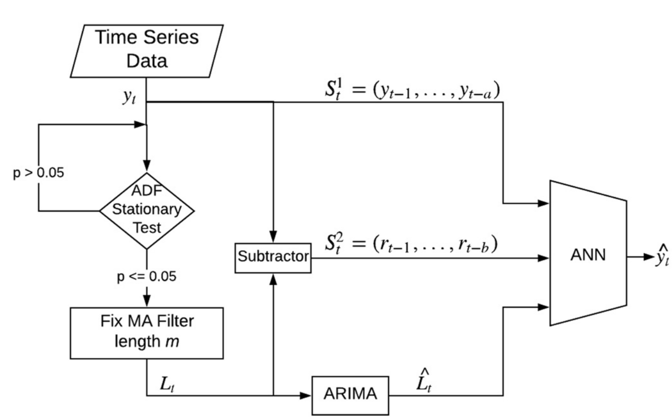


Fig. 1. Proposed hybrid method.

The more negative ADF test result means the stronger rejection of the existence of unit root for a given time series. Therefore, a negative ADF result implies that the given dataset is stationary. The well accepted threshold is 0.05 which is also used in this study to adjust MA filter length.

After the linear component is achieved with MA filter, a linear model is constructed as shown in Eq. (7). The stationary component l is modeled as a linear function of past values of the data series $l_{t-1}, l_{t-2}, \dots, l_{t-p}$ and random error series $\epsilon_{t-1}, \epsilon_{t-2}, \dots, \epsilon_{t-q}$ in Eq. (1) using ARIMA model.

$$\hat{L_t} = g(l_{t-1}, l_{t-2}, \dots, l_{t-p}, \epsilon_{t-1}, \epsilon_{t-2}, \dots, \epsilon_{t-q}) \tag{7}$$

where g is a linear function of ARIMA.

Finally, nonlinear modeling ANN is used to implement functional relationship between components as indicated in Eq. (4). The past observed data $y_{t-1}, y_{t-2}, \dots, y_{t-a}$, present ARIMA forecast result of the decomposed stationary data $\hat{L_t}$, and residuals of the data decomposition $r_{t-1}, r_{t-2}, \dots, r_{t-b}$ are fed to ANN as indicated in Eq. (8):

$$\begin{aligned}
S_t^1 &= (y_{t-1}, y_{t-2}, \dots, y_{t-a}) \\
S_t^2 &= (r_{t-1}, r_{t-2}, \dots, r_{t-b}) \\
\hat{y_t} &= f(S_t^1, \hat{L_t}, S_t^2) \\
\hat{y_t} &= f(y_{t-1}, y_{t-2}, y_{t-a}, \hat{L_t}, r_{t-1}, r_{t-2}, \dots, r_{t-b})
\end{aligned} \tag{8}$$

where f is the nonlinear function of ANN, a and b are parameters of the model which show how much we will go back in time to use as features to ANN. Time series data determines how many of those features in the residual path and observed data path are going to be used in the nonlinear model. For example, if the given data does not show volatility, then the residual variable b in Eq. (8) might come out even as zero in tuning process. Likewise, a variable in Eq. (8) is also empirically determined in the tuning process.

The proposed model does not only exploit the unique strength of single models, but also eliminates the three strong assumptions performed by other hybrid methods. Therefore, risk of low forecasting performance in unexpected situations is highly avoided. The competitive performance of our proposed algorithm is shown in our experimental results by using various type of datasets.

4. Empirical results

The performance results of the proposed hybrid method along with the other methods discussed in this paper are evaluated on four different datasets. Three of them are well-known benchmark datasets - the Wolf's sunspot data, the Canadian lynx data, and the British pound/US dollar exchange rate data – which have been widely used in statistics and the neural network literature [13,22,23,25]. The other dataset is publicly available electricity price of Turkey Intraday Market [29]. These four different time series data are originated from different disciplines and show different characteristics. While one dataset (e.g. GbpUsd) is highly non-stationary and volatile, another one (e.g. Lynx) shows linear behaviors. Similarly, while one of them includes seasonality (e.g. Sunspot), the remaining dataset(e.g. Intraday) does not. This diversity has enabled us to make our experiments with a wide spectrum of datasets.

In the experiments, only one-step-ahead forecasting is considered. In order to compare accuracy performances, three evaluation metrics are used: Mean Absolute Error (MAE), Mean Squared Error (MSE) and Mean Absolute Scaled Error (MASE) whose formulations are indicated as follows respectively:

$$\begin{aligned}
\text{MAE} &= \frac{1}{n}\sum_{t=1}^{n} |e_t| \\
\text{MSE} &= \frac{1}{n}\sum_{t=1}^{n} e_t^2 \\
\text{MASE} &= \frac{n-1}{n}\frac{\sum_{t=1}^{n} |e_t|}{\sum_{t=2}^{n} |y_t - y_{t-1}|}
\end{aligned} \tag{9}$$

where $e_t = y_t - \hat{y}_t$ and y_t is the actual data value, $\hat{y}_t$ is the forecasted value at given time t. While MAE specifies the average of the absolute errors over the performed prediction, MSE measures the average of the squared error. Since prediction errors are squared in MSE metric, it gives relatively greater influence to larger errors. Hence, this property makes MSE useful when large errors are not desired. On the other hand, MAE does not penalize outliers like MSE, since it equally weights prediction errors in the average. Both metrics are good when comparing algorithms for a dataset. However, when comparing an algorithm for time series which have different scales, they might not be reliable, since both MAE and MSE results depend on the scale of the given data. Therefore, a scale-free error metric MASE can be used to compare forecast accuracy between scaled series.

To assess the forecasting performance of the different methods, each dataset is divided into training and testing sets. While the training data is used for model development, the test data is used to evaluate the established model. In order to tune the hyperparameters the methods, the last 20% of training dataset is used as validation set. While choosing the right ANN model, since there is not an accepted method for network configuration, an empirical approach is performed. In this approach, ANN is methodically build for each combination of algorithm parameters and evaluated on the validation set. The achieved best configuration is used as a final model.

In addition, due to the fact that ANN performs random initialization and produces different results at each run, the methods which include ANN algorithm are executed 50 times and average results are reported. This number is not a hyperparameter to tune but high enough to get reliable and robust results. Table 1 gives the forecasting results of all examined methods on these all four datasets. The table also presents the results of the naïve method in order to have a baseline comparison. This method ($y_t = y_{t-1}$) uses the value at the previous time step $t-1$ to predict the expected outcome at the current time step t.

In order to assess whether the average results of the methods are statistically different from each other, t-test analysis is used [30]. As a result of the test, p-values are obtained for pairwise comparison between our method's and the other methods' error metrics. If p-value is less than 0.05, it can be accepted as our method is statistically different from the compared method in the test [30]. Table 2 presents the p-values for four pairwise comparison for all datasets. Since MASE is the scaled up of MAE, the test results in two metrics are same. To avoid the repetition instead of two of them, the results for only MAE are reported. Table 2 shows that the proposed method is significantly different from all examined methods for all experimented datasets in MSE metric. In MAE metric, the proposed methods is statistically significant than at least three of the examined methods.

4.1. Forecasts for sunspot dataset

The Wolf's sunspot series, which contains annual activity of spots visible on the face of the sun, has been extensively used in numerous linear and nonlinear models [22]. The data includes the annual count of sunspots from 1700 to 1987 (see Fig. 2) giving a total of 288 observations. ADF stationarity test result of the

Table 1
Performance comparison for all datasets.

Datasets	Methods metrics	Naïve Method	ANN	ARIMA	Zhang's method	K-B's method	B-R's method	Proposed method
*Sunspot	*MAE*	22.96	14.23 ± 0.40	13.37	13.14 ± 0.34	10.62 ± 0.26	11.39 ± 0.22	**10.48 ± 0.27**
	MSE	920.73	353.12 ± 24.26	306.97	289.31 ± 23.01	205.08 ± 15.06	239.90 ± 22.89	**194.29 ± 13.36**
	MASE	1.01	0.62 ± 0.01	0.59	0.58 ± 0.01	0.47 ± 0.01	0.50 ± 0.01	**0.46 ± 0.01**
Lynx	*MAE*	0.2308	0.1249 ± 0.0029	0.1198	**0.1003 ± 0.0023**	0.1025 ± 0.0025	0.1102 ± 0.0028	0.1013 ± 0.0031
	MSE	0.0687	0.0241 ± 0.0019	0.0231	0.0173 ± 0.0016	0.0175 ± 0.0016	0.0189 ± 0.0019	**0.0162 ± 0.0018**
	MASE	1.1425	0.6185 ± 0.0144	0.5932	**0.4966 ± 0.0113**	0.50757 ± 0.0127	0.5457 ± 0.0138	0.5016 ± 0.0157
Gbp/Usd	*MAE*	512.41	428.55 ± 60.45	435.72	429.52 ± 48.54	406.22 ± 24.65	436.34 ± 36.87	**404.90 ± 23.43**
	MSE	3.97	3.47 ± 0.37	3.52	3.45 ± 0.30	3.10 ± 0.22	3.50 ± 0.33	**2.95 ± 0.26**
	MASE	1.297	1.085 ± 0.153	1.103	1.087 ± 0.122	1.028 ± 0.062	1.104 ± 0.145	**1.025 ± 0.059**
Intraday	*MAE*	21.54	20.10 ± 1.242	20.22	19.16 ± 0.925	19.79 ± 1.708	19.50 ± 0.981	**18.81 ± 0.972**
	MSE	797.03	617.46 ± 37.17	652.72	594.09 ± 27.10	600.93 ± 46.75	619.67 ± 37.07	**581.38 ± 34.77**
	MASE	1.12	1.04 ± 0.06	1.05	0.99 ± 0.04	1.03 ± 0.08	1.01 ± 0.05	**0.98 ± 0.05**

* Khashei–Bijari's method is shown as K-B's method. ** Babu–Reddy's method is shown as B-R's method. *** MAE and MSE results are multiplied with 10^{-5} in Gbp/Usd dataset **** The values shown with ± give the standard deviation values.

Table 2
p-value test result comparisons of the proposed method.

Datasets	Methods metrics	ANN	Zhang's method	Khashei–Bijari's method	Babu–Reddy's method
Sunspot	*MAE*	0.001	0.001	0.007	0.001
	MSE	0.001	0.001	0.001	0.001
Lynx	*MAE*	0.001	**0.135**	0.038	0.001
	MSE	0.001	0.002	0.011	0.001
Gbp/Usd	*MAE*	0.017	0.022	**0.124**	0.037
	MSE	0.001	0.001	0.036	0.001
Intraday	*MAE*	0.001	0.012	0.001	0.001
	MSE	0.001	0.049	0.017	0.001

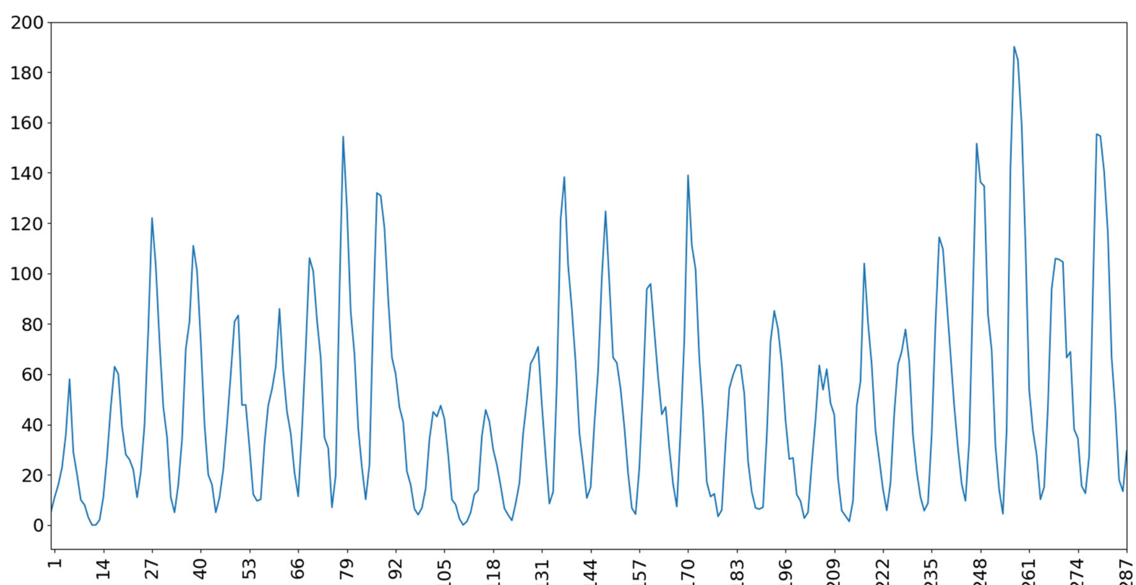


Fig. 2. Sunspot series (1700–1987).

dataset is 0.083 which is greater than the threshold 0.05. This implies that there is a unit root on the dataset, thus the dataset can be regarded as non-stationary time series. 288 observations in the dataset is divided into two samples: 221 observations between 1700–1920 years are considered as training data to develop the model, the last 67 observations between 1921–1987 years are considered as test data and used to evaluate the model performance.

In Sunspot dataset, when ARIMA is individually used as a forecasting method, we also set the order of ARIMA to 9 (AR(9)) as same as the other many studies [13,22,23]. When ANN is individually used as a forecasting method, similar to these studies, three layered $4 \times 4 \times 1$ ANN architecture is used which is composed of four input nodes, four hidden nodes, and one output node.

In the proposed method, the linear component comes out when the MA filter length is 15. After using the filter, the achieved linear component has 0.006 stationary test result which indicates its stationarity, since it is a value less than the threshold, 0.05. The best fitted neural network in the final step of the proposed hybrid method has 7 nodes in the input layer where 4 of them are observed values, 2 of them are residuals, and one node is assigned for the result of linear component forecast. According to our experiments, when the number of hidden nodes are adjusted to same number of nodes as in the input layer, the best fitted ANN model is achieved.

When numerical results of Sunspot dataset given in Table 1 are analyzed, individual methods such as ARIMA, ANN have apparently

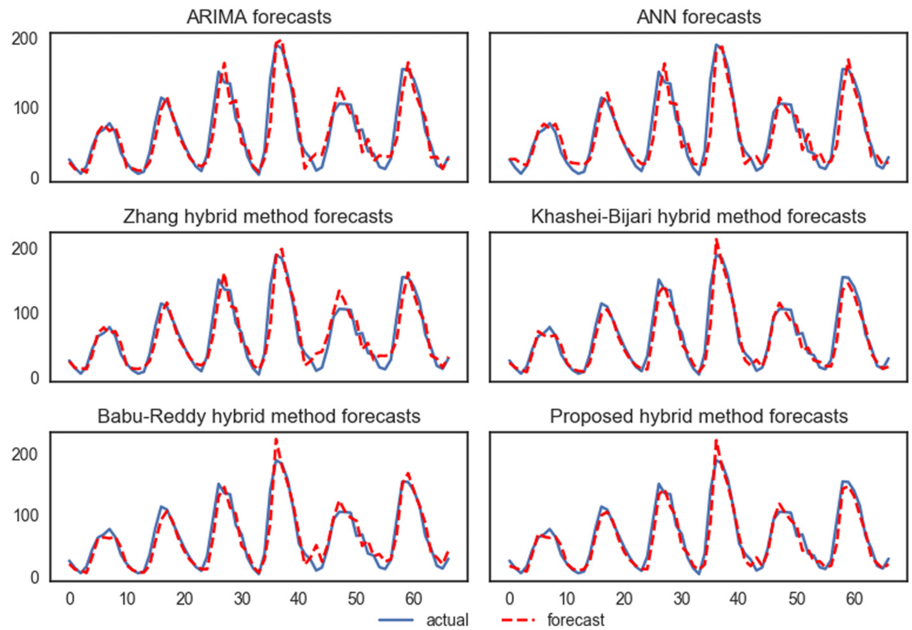


Fig. 3. Sunspot data forecasts using various methods.

lowest performance as compared to hybrid methods. This suggests that either ARIMA or ANN, when individually used, do not capture all patterns in the data series. Therefore, combining two methods by taking advantage of each of them can be an effective way to overcome this limitation. We indeed observe in Table 1 that the hybrid methods Zhang, Khashei–Bijari, and Babu–Reddy methods produce better results as compared to individual ones. However, they produce lower forecasting performance than our proposed hybrid method. The assumptions those hybrid methods make can be restricting in many situations as mentioned in Section 3. Our proposed hybrid method eliminates those assumptions and yields better generalization performance. The comparison of the actual and forecast values for all examined method are given in Fig. 3.

4.2. Forecasts for Lynx dataset

The lynx dataset, which contains the number of lynx trapped per year in the Mackenzie River district of Northern Canada, is an another extensively analyzed time series data in the literature [13,22]. The data shows a periodicity of approximately 10 years as seen in Fig. 4. Moreover, ADF stationary test results of the dataset is 0.056 which implies that the dataset is almost stationary. There are 114 observations in the data, corresponding to the period of 1821–1934. The first 100 observations between 1821–1920 years are considered as training data to develop the model, the last 14 observations between 1921–1934 years are used as test data to evaluate the model performance. In addition, like in other studies [13,22], the logarithms (to the base 10) of the data are used in the analysis.

When ARIMA is used as an individual model, we used the AR model of order 12 (AR(12)) for Lynx dataset which is also used by [13,22]. Similar to these studies, three layered $7 \times 5 \times 1$ ANN architecture is used when ANN is individually used as a forecasting method.

In the proposed method, the linear component is extracted from the Lynx dataset when the MA filter length is 5. The relatively short MA filter length was expected, since the ADF test result shows a certain level of stationarity in the data. As a result of MA filter, the achieved linear component has 0.006 stationary test result which indicates even more stationarity to be properly modeled by ARIMA. The best fitted neural network in the final step of the proposed hybrid method has 9 nodes in the input layer where 5 of them are observed values, 3 of them are residuals, and one node is assigned for the result of linear component forecast. According to our tuning experiments, when the number of hidden nodes are adjusted to the same number of nodes as in the input layer, the best fitted ANN model is achieved.

In this dataset, among the individually used methods, ARIMA gives better accuracy as compared to ANN in contrast to the Sunspot dataset (see in Table 1). This is most likely due to the fact that Lynx dataset is more stationary dataset compared to Sunspot dataset. Due to this relative stationarity, the effect of hybrid methods might not be easily observed. Although we have circumstances which do not necessarily favor the data decomposition and model combination, hybrid methods do not give a lower performance than the individual linear models, and they even provide better results. The highest performance is mostly achieved by our proposed hybrid method. Fig. 5 compares the actual and forecast values for all examined methods.

4.3. Forecasts for Gbp/Usd dataset

The other benchmark dataset is the exchange rate between British pound and US dollar which contains weekly observation from 1980 to 1993, giving 731 data points in the time series. Predicting exchange rate is an important yet difficult task due to high volatility. ADF stationary test result of the dataset is 0.58 which is highly greater than the threshold 0.05. This implies that the dataset is highly volatile and non-stationary. This non-stationarity can be even seen in the plot, given in Fig. 6, which shows numerous changing turning points in the series. Similar to other datasets, the experimental setup is same as in previous hybrid studies [13,22] where data is transformed using natural logarithmic function and separated into two samples. The first 679 observations from 1980 to 1992 years are considered as training data to develop the model, the last 52 observations between 1992–1993 years are used as test data to evaluate the model performance.

In this dataset, when ARIMA is individually used, rather than using regression type of model in the ARIMA itself, random walk model is chosen as best-fitted ARIMA model. This approach has

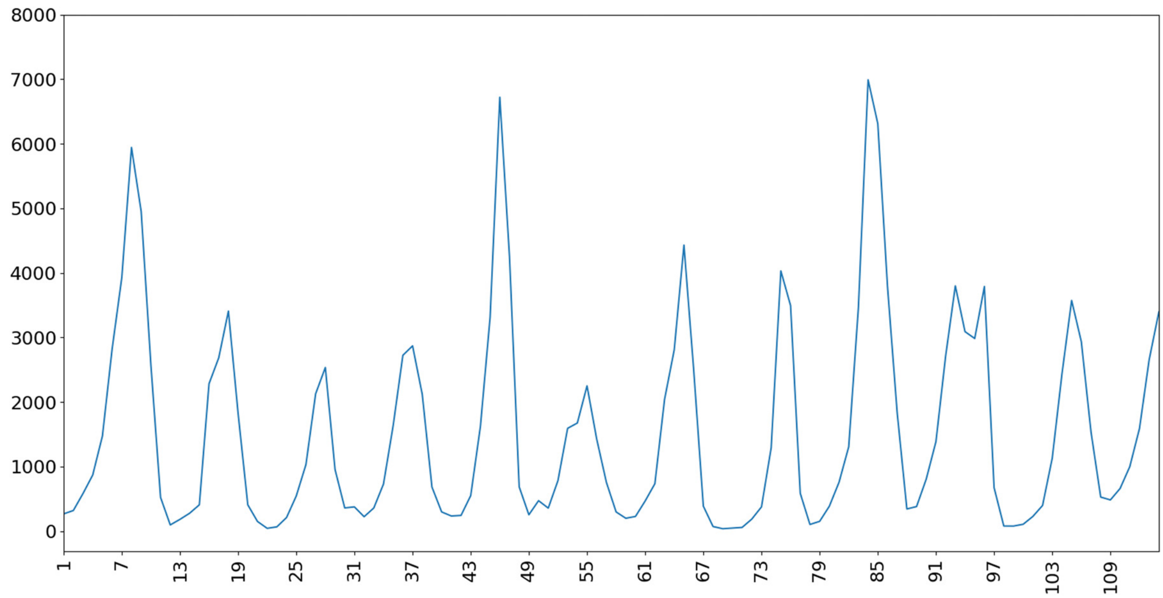


Fig. 4. Canadian lynx data series (1821–1934).

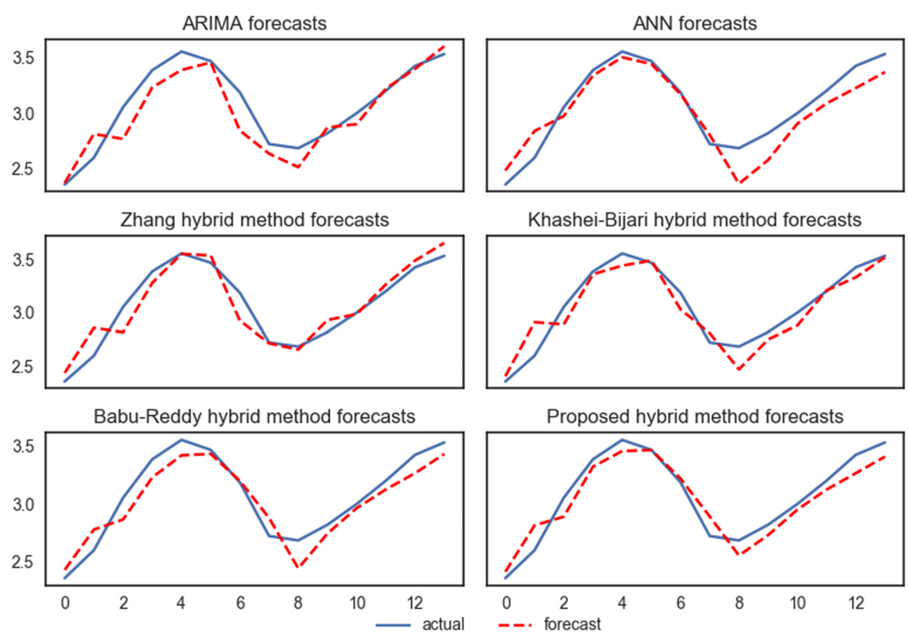


Fig. 5. Lynx data forecasts using various methods.

been used by Zhang [13] and also been suggested by many studies in the exchange rate literature [31]. In this model, the most recent observation is the best guide for the next forecast. When ANN is individually used as a forecasting method, the best fitted ANN is set as three layered $7 \times 6 \times 1$ architecture.

In the proposed method, in order to decompose this highly volatile data, MA filter length comes out to be 40. As a result of this decomposition, ADF test result of the obtained component is 0.007 which indicates the stationarity of the component. To compute the best final forecast in the proposed model, the ANN is constructed as three layered $9 \times 9 \times 1$ architecture. In this architecture, input layer is composed of the last 5 of observed values, the last 3 of residuals, and the result of linear component forecast.

Results of the Gbp/Usd dataset forecasts are compared in Fig. 7. Both ANN and hybrid methods have much better performance than the individual ARIMA method for a highly fluctuating forecast horizon. The proposed hybrid method is able to capture this volatile pattern much better and outperforms the other methods in all error metrics.

4.4. Forecasts for intraday dataset

The last analyzed dataset is Turkey intraday electricity market price data which is publicly available [29]. The dataset contains 581 observations which consist of daily averaged prices from July 2015 to December 2017. As compared to datasets such as Sunspot and Lynx, data pattern in this dataset is also highly fluctuating (see Fig. 8). ADF stationary test result of the dataset shows 0.27 value which is highly greater than the threshold of 0.05. Natural logarithmic transformation is applied on the dataset for scaling purposes.

In this dataset, when ARIMA is individually used as a forecasting method, we found autoregressive model of order 9 (AR(9)) to

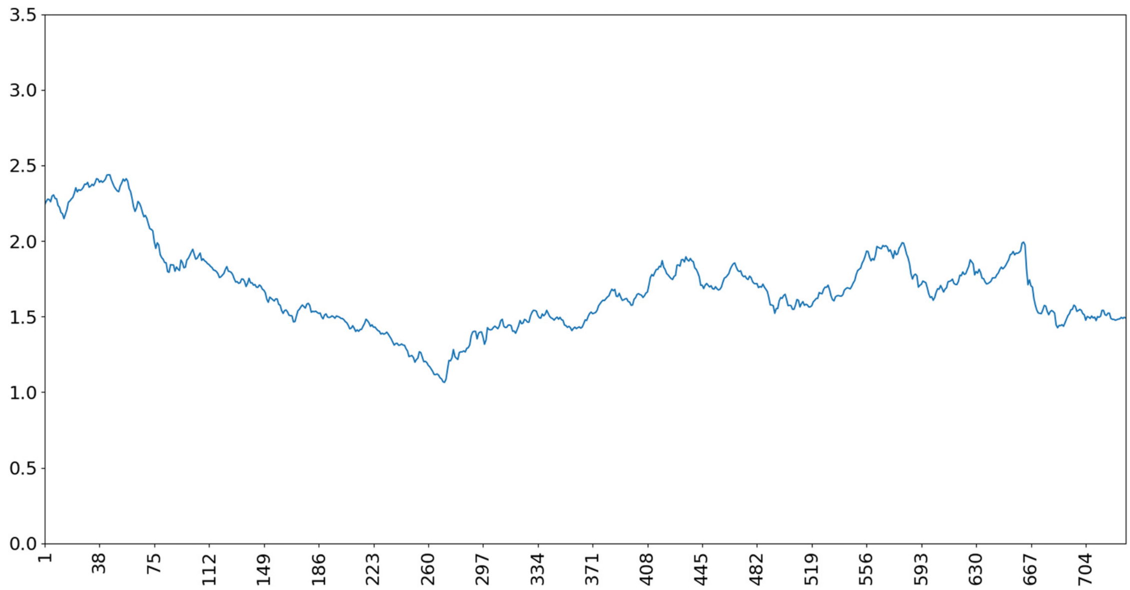


Fig. 6. Weekly British pound/US dollar exchange rate series (1980–1993).

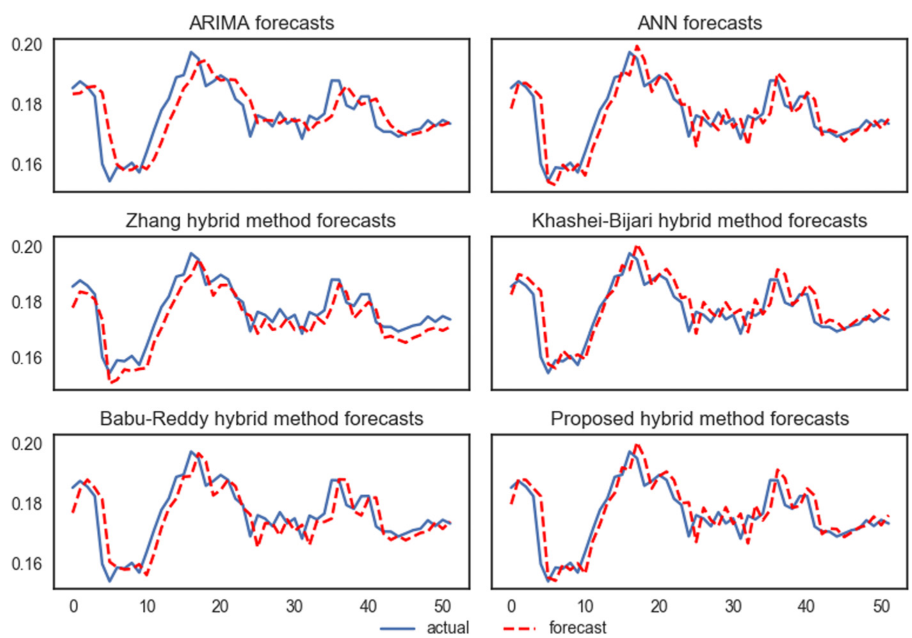


Fig. 7. Gbp/Usd data forecasts using various methods.

be the most parsimonious one among all ARIMA models. The best fitted individual ANN model is achieved in three layered $3 \times 6 \times 1$ architecture, after our tuning process.

In this dataset, the length of MA filter which we use in the proposed hybrid method comes out as 6. The obtained linear component after the MA filter has 0.004 ADF stationary test result; this implies that the component can be properly modeled by ARIMA method in our architecture. The best fitted ANN in the final step of the proposed method has $17 \times 17 \times 1$ architecture. Input layer is composed of the last 8 of original data, the last 8 of residuals and the result of linear component forecast.

Similar to Gbp/Usd dataset, Intraday dataset is highly volatile and non-stationary which cannot be effectively modeled by using only a linear model. As can be seen in Table 1, ANN and hybrid methods significantly outperform the individual linear model of ARIMA. Furthermore, the proposed method gives remarkably superior accuracy as compared to other hybrid methods in all error metrics. The comparison of the actual and forecast values for all examined method are given in Fig. 9.

5. Discussion and improvement

There are several important results obtained in our experiments Firstly, when individual methods' results are compared among themselves, we see that ARIMA outperforms ANN for the datasets which present more linearity and vice versa (see Table 1). Moreover, hybrid methods have better performance as compared to individual ones especially in more fluctuating datasets. Furthermore, the assumptions made by other hybrid methods degenerate the forecasting performance when unexpected situations occur in the data. For example, Zhang's and Babu–Reddy's methods assume that the relationship between linear and nonlinear components

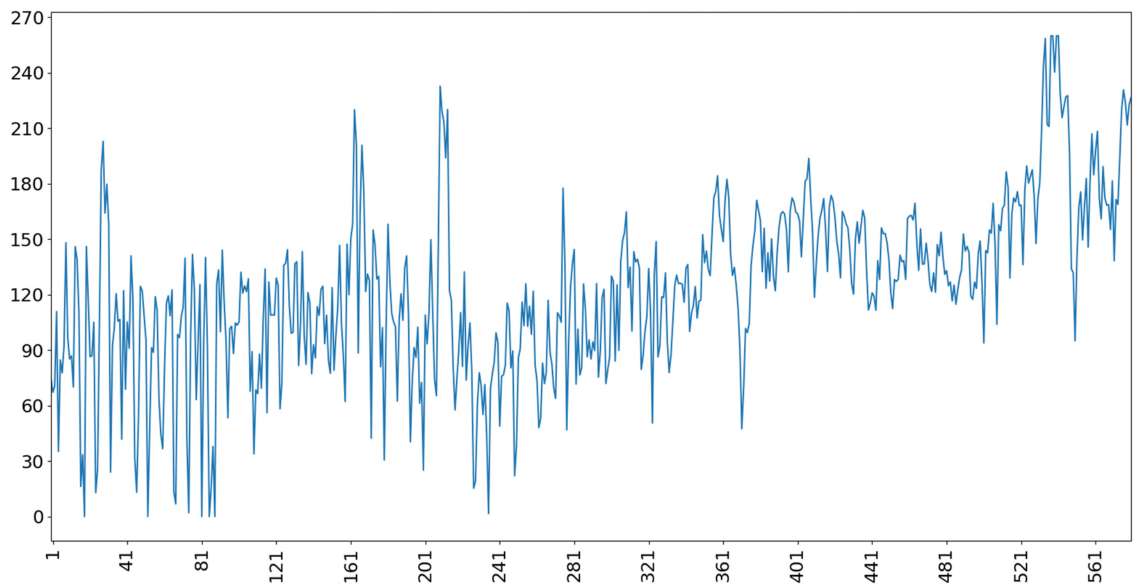


Fig. 8. Electricity price of intraday market in Turkey (Jul. 2015–Dec. 2017).

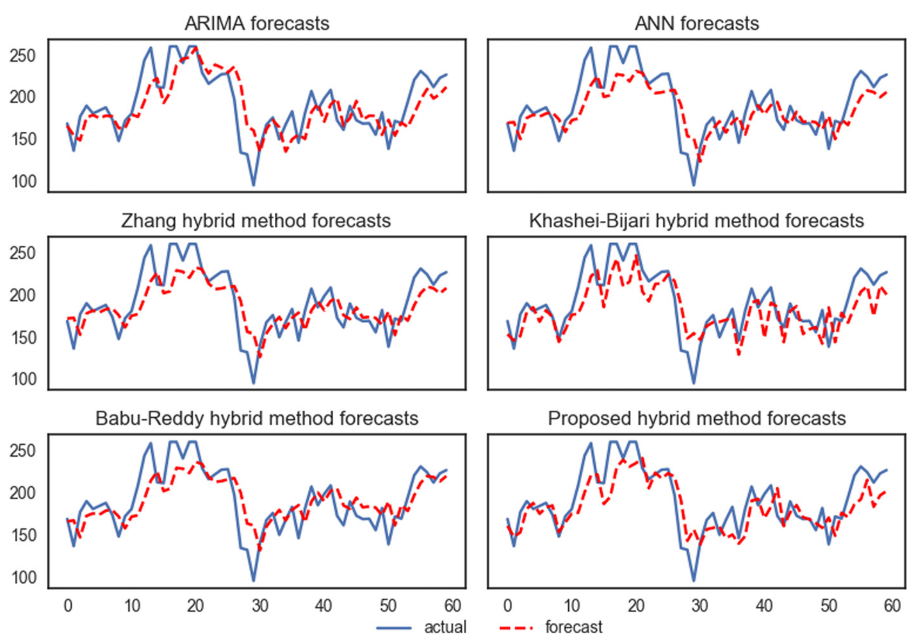


Fig. 9. Intraday data forecasts using various methods.

is additive. If linear and nonlinear components are not additively associated and the relation is different (i.e., it can be multiplicative), the possible complex relationship between components might be overlooked and the forecasting performance might be degenerated. Another assumption that might not always hold is that residuals might not show nonlinear pattern property. Additionally, Zhang's and Khashei–Bijari's methods do not actually decompose data into linear and nonlinear components, but they assume that linear component can be extracted by ARIMA and error sequences shows nonlinear pattern. As a result; such assumptions may lead to low forecasting performances when unexpected scenarios happen. Our proposed hybrid method avoids making those assumptions and creates more general models and outperforms the other examined methods.

Fig. 10 compares the distribution of MASE results of datasets with ADF test results of the corresponding dataset. In this figure, boxplots are drawn by using MASE values of all examined methods for each dataset, presented in Table 1. When we compare error results among time series data by using a scale-invariant error metric MASE, it is observed that the more non-stationarity in a dataset leads to a higher error value. For example, Lynx dataset, which turns out to be the most linear among all datasets according to ADF test results, has the lowest MASE results. On the other hand, Gbp/Usd dataset, which shows the most nonlinearity according to ADF test results, has the highest MASE results. As a result of this, we can conclude that having more regular data distribution in a time series leads to more accurate results in forecasting. This conclusion motivates us to propose an improvement on our already best performing proposed hybrid method. This improved method aims to produce more stationary subseries from given time series by using a multi-scale decomposition technique. Then, those achieved linear subseries can be modeled with a higher accuracy using the proposed hybrid method.

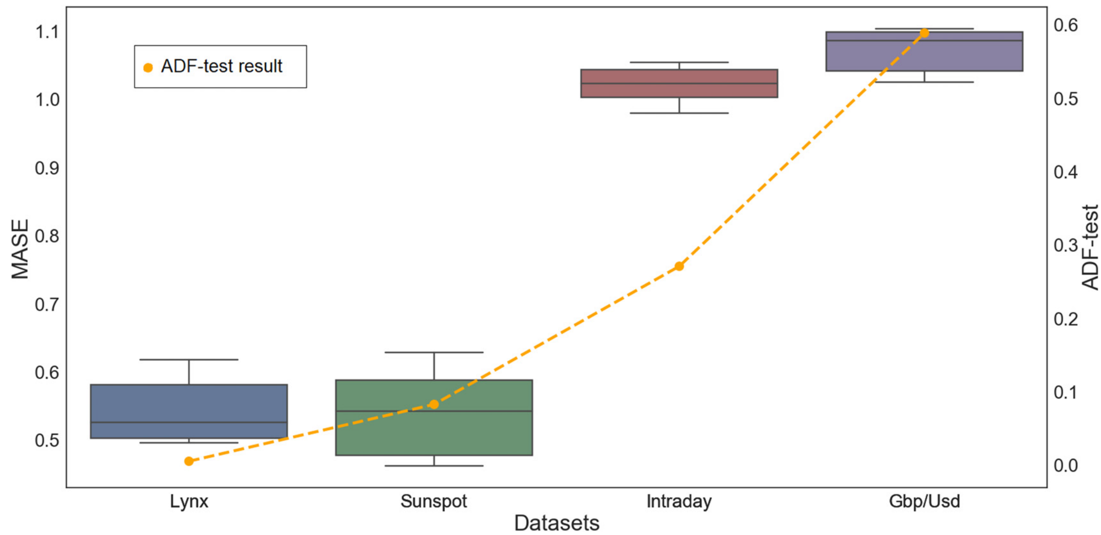


Fig. 10. Comparison of MASE distribution with ADF test results for all datasets.

In the literature there are several multi-scale decomposition methods such as Empirical Mode Decomposition (EMD), Wavelet Packet Decomposition (WPD), Fourier Transform (FT) and etc. [32]. Since EMD does not make a priori assumption about the given time series and preserves time scale of the data throughout the decomposition, it is a more preferable technique than the others for decomposing time series [24]. In addition, it is shown that the combination of EMD and traditional hybrid methods give promising results [25].

The main principle of EMD is to decompose a given time series data into a sum of several subseries. Those subseries are called *Intrinsic Mode Functions (IMFs)* and the remaining component after subtracting the summation of IMFs from the original data is called *residue*. These subseries have two important properties which allow them to be easily modeled: Each subseries has its own local characteristic time scale and they are relatively stationary subseries. Let $y(t)$ be a given time series data, and then the EMD calculation can be described as follows:

$$y(t) = \sum_{i=1}^{n} IMF_i(t) + R_n(t) \tag{10}$$

where $IMF_i(t)$ $(i = 1, 2, \ldots, n)$ represents the different subseries, and $R_n(t)$ is the residue after summation of n IMFs are subtracted from the original data.

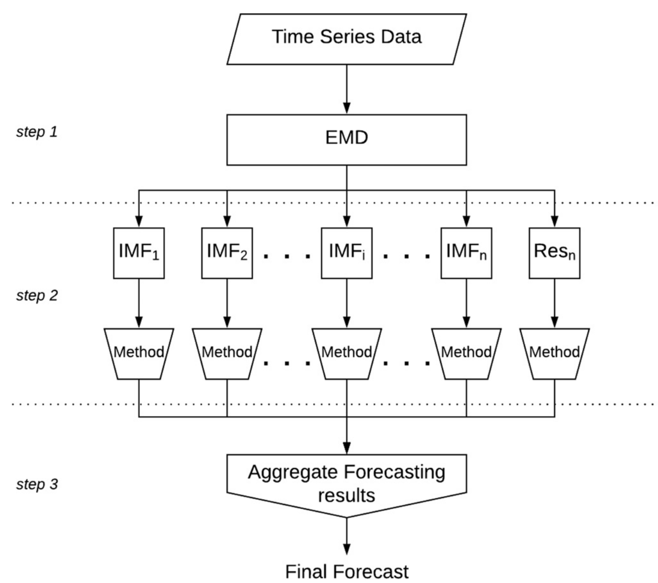


Fig. 11. The hybrid architecture using EMD.

The EMD-based methods includes three main steps, as seen in Fig. 11: In the first step, the original time series data is decomposed into IMFs. In the second step, forecasting is performed by using our hybrid method on each IMF. In the last step, forecast results of each individual model are summed up to achieve the final forecast of the original time series. We use an additive function in the end to capture the additive relation of IMFs with the original data.

The EMD-based methods are evaluated on the same datasets by using the experimental setup in Fig. 11. In order to evaluate the effect of EMD, all examined methods are executed in the second step of the algorithm (see Fig. 11). Table 3 gives the forecasting results of all examined methods with EMD on four all datasets. When these results are compared with the previous ones showed in Table 1, the methods with EMD give significantly higher accuracies. The percentage improvement for all datasets are presented in Table 5. The improvements at each dataset varies between 23% and 89% for all error metrics. We also provide a bar chart (see Fig. 12) that shows the MASE results for each method which are averaged over the results of all datasets. The chart indicates that methods with EMD achieve remarkably less error in their forecasts. Further analysis shows that EMD-based methods give even greater improvements in the accuracies for non-stationary datasets. This is due to the fact that EMD is able to resolve nonstationarity and alleviate high volatility problem in time series data [25]. For example, while our hybrid method with EMD improves MASE results of relatively stationary datasets Sunspot and Lynx 30% and 28% respectively, this improvement jumps over 50% in Gbp/Usd and Intraday datasets which are highly non-stationary (see Table 5). Another indication of EMD's capability of solving volatility problem is that ARIMA is able to achieve better results than ANN for all datasets (see Table 5), although ANN without using EMD was better in non-stationary datasets (see Table 1).

The t-test anlaysis of the experimental results reported in Table 3 are presented in Table 4 for all datasets and error metrics. As Table 4 suggest, the proposed method using EMD is statistically significant than at least three of the examined methods for all experimented datasets.

Table 3
Performance comparison of methods using EMD for all datasets.

Datasets	Methods metrics	ANN	ARIMA	Zhang's method	K-B's method	B-R's method	Proposed method
Sunspot	*MAE*	8.33 ± 0.58	7.72	7.46 ± 0.57	7.76 ± 0.48	7.92 ± 0.52	**7.28 ± 0.47**
	MSE	120.14 ± 9.80	99.07	90.04 ± 8.47	99.17 ± 11.5	100.64 ± 9.20	**87.86 ± 9.11**
	MASE	0.368 ± 0.025	0.341	0.330 ± 0.025	0.343 ± 0.014	0.350 ± 0.023	**0.322 ± 0.020**
Lynx	*MAE*	0.0912 ± 0.0046	0.0772	0.0751 ± 0.0034	0.0782 ± 0.0043	0.0788 ± 0.0045	**0.0760 ± 0.0035**
	MSE	0.0131 ± 0.0010	0.0099	0.0099 ± 0.0009	0.0099 ± 0.0010	0.0101 ± 0.0008	**0.0092 ± 0.0008**
	MASE	0.4516 ± 0.0230	0.3822	0.3718 ± 0.0168	0.3872 ± 0.0213	0.3902 ± 0.0226	**0.3763 ± 0.0176**
Gbp/Usd	*MAE*	190.08 ± 8.25	146.03	142.92 ± 6.13	146.11 ± 7.66	147.18 ± 9.87	**141.38 ± 6.87**
	MSE	0.5610 ± 0.0408	0.3578	0.3479 ± 0.0271	0.3581 ± 0.0319	0.3633 ± 0.0292	**0.3285 ± 0.0280**
	MASE	0.4812 ± 0.0209	0.3697	0.3618 ± 0.0155	0.3699 ± 0.0194	0.3726 ± 0.0250	**0.3579 ± 0.0174**
Intraday	*MAE*	10.96 ± 1.39	10.79	10.37 ± 1.36	9.88 ± 1.03	10.05 ± 1.20	**8.93 ± 1.04**
	MSE	182.86 ± 19.60	158.64	158.88 ± 17.60	162.55 ± 15.24	165.41 ± 16.33	**130.79 ± 14.91**
	MASE	0.57 ± 0.07	0.56	0.54 ± 0.07	0.51 ± 0.05	0.52 ± 0.06	**0.45 ± 0.05**

* Khashei–Bijari's method is shown as K-B's method. ** Babu–Reddy's method is shown as B-R's method. *** MAE and MSE results are multiplied with 10^{-5} in Gbp/Usd dataset **** The values shown with ± give the standard deviation values.

Table 4
p-value test result comparisons of the proposed method using EMD.

Datasets	Methods metrics	ANN	Zhang's method	Khashei–Bijari's method	Babu–Reddy's method
Sunspot	*MAE*	0.001	0.016	0.001	0.001
	MSE	0.001	**0.233**	0.001	0.001
Lynx	*MAE*	0.001	**0.136**	0.001	0.002
	MSE	0.001	0.001	0.001	0.001
Gbp/Usd	*MAE*	0.001	**0.328**	0.011	0.001
	MSE	0.001	0.004	0.003	0.001
Intraday	*MAE*	0.001	0.001	0.001	0.001
	MSE	0.001	0.001	0.001	0.001

Table 5
Percentage improvement in the methods when EMD is used.

Datasets	Methods metrics	ANN (%)	ARIMA (%)	Zhang's method (%)	Khashei–Bijari's method (%)	Babu–Reddy's method (%)	Proposed method (%)
Sunspot	*MAE*	41.1	42.2	43.2	26.9	30.4	30.5
	MSE	65.9	67.7	68.8	51.6	58.0	54.7
	MASE	41.1	42.2	43.2	26.9	30.4	30.5
Lynx	*MAE*	26.9	35.5	25.1	23.7	28.4	28.4
	MSE	45.6	57.1	42.7	43.4	46.5	43.2
	MASE	26.9	35.5	25.1	23.7	28.4	28.4
Gbp/Usd	*MAE*	55.6	66.4	62.3	64.0	66.2	65.0
	MSE	83.8	90.0	90.1	88.7	89.7	89.1
	MASE	55.6	66.4	62.3	64.0	66.2	65.0
Intraday	*MAE*	45.4	46.6	44.4	50.0	48.4	52.5
	MSE	70.3	75.6	73.2	72.9	73.3	77.5
	MASE	45.4	46.6	44.4	50.0	48.4	52.5

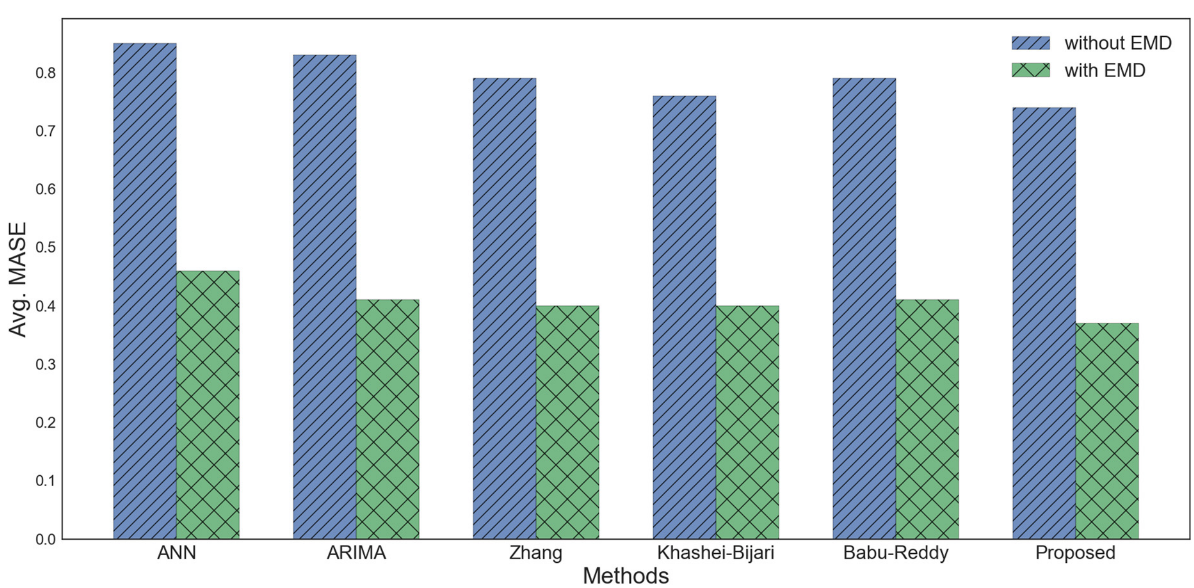


Fig. 12. Average MASE results of the methods for all datasets with/without using EMD.

In final, we want to point out that our proposed hybrid method with EMD gives the best results as compared to other methods (see Table 5). Subseries obtained from EMD are relatively stationary as compared to original data however, they still show fluctuations in their frequency range. Performing one more decomposition on these subseries using MA filter and constituting functional relation between the stationary part, residuals, and the original data values outperforms all other remaining methods.

6. Conclusions

Time series forecasting is an important yet often a challenging task used in many different application domains. The studies in the literature mainly focus on either linear or nonlinear modelings individually or a combination of them. While linear models such as ARIMA gives better forecasting accuracy with stationary time series data, nonlinear methods such as ANN is more appropriate for non-stationary datasets. In order to take advantage of the unique strength of each different type of methods in a more general setting, hybrid methods are proposed. Hybrid methods basically use linear and nonlinear modeling, ARIMA and ANN respectively on the corresponding decomposed components and then combine the results. Hybrid ARIMA-ANN methods give better results in general as compared to the cases where they are individually used. However, they generally suffer from the assumptions they make while constructing their model. These assumptions lead to produce inconsistent results and give low accuracies in overall if unexpected situations occur.

In this study, a new hybrid ARIMA-ANN model based forecasting method is proposed to overcome three main assumptions made by traditional hybrid ARIMA-ANN models. Firstly, the proposed method removes the assumption that the linear component is the ARIMA model output of the given data. Rather, it extracts the linear component by using MA filter. It is known that data showing linear characteristics can be more accurately modeled by linear methods. Therefore, properly decomposed data yields more accurate linear forecasting and consequently more accurate final results in the hybrid methods. Secondly, the proposed method does not directly model residuals via a nonlinear method ANN, since the assumption of that residuals might comprise valid nonlinear patterns, does not always hold. Thirdly, the proposed method does not restrict linear and nonlinear component modeling and also combining the results of them. Rather, it can capture structures of the linear and nonlinear components in a better way, and produces more general models than those existing hybrid models.

In the light of our experimental results, we can conclude that forecasting performance gets better if more stationary time series data is provided. This result motives us to make original time series data more stationary in order to improve accuracy results. We show that when EMD multi-scale data decomposition is combined with all examined methods, accuracy results can be remarkably improved. Our experimental results indicate that our hybrid method with EMD gives remarkably superior accuracy as compared to all other examined methods.

As a future work, further time series analysis methods can be applied to each EMD component as a pre-processing step to choose the most proper method to apply. In addition, rather than one-step-ahead forecasting that we targeted in this study, the proposed method could be adopted and enhanced to multi-step-ahead forecasting.

Declaration of interests

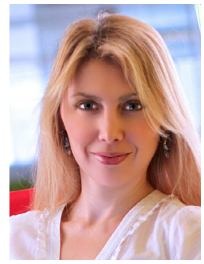
None.

References

[1] J.C. Brocklebank, D.A. Dickey, B. Choi, SAS for Forecasting Time Series, 3rd Ed., SAS Institute, 2018.
[2] J. Contreras, R. Espinola, F. Nogales, A. J. Conejo, ARIMA models to predict next-day electricity prices, IEEE Power Eng. Rev. 22 (2002) 57, doi:10.1109/MPER.2002.4312577.
[3] E. Gonzalez-Romera, M.A. Jaramillo-Moran, D. Carmona-Fernandez, Monthly electric energy demand forecasting based on trend extraction, IEEE Trans. Power Syst. 21 (4) (2006) 1946–1953, doi:10.1109/TPWRS.2006.883666.
[4] K.P. Vishwajith, P.K. Sahu, B.S. Dhekale, P. Mishra, Modelling and forecasting sugarcane and sugar production in india, Indian J. Econ. Dev. 12 (2016) 71, doi:10.5958/2322-0430.2016.00009.3.
[5] W.-c. Wang, K.-w. Chau, D.-M. Xu, X.-Y. Chen, Improving forecasting accuracy of annual runoff time series using ARIMA based on EEMD decomposition, Water Resour. Manag. 29 (2015) 2655–2675, doi:10.1007/s11269-015-0962-6.
[6] A. Lapedes, R. Farber, Nonlinear signal processing using neural networks: prediction and system modelling, in: Proceedings of the IEEE International Conference on Neural Networks, San Diego, 1987.
[7] W.-S. Chen, Y.-K. Du, Using neural networks and data mining techniques for the financial distress prediction model, Expert Syst. Appl. 36 (2) (2009) 4075–4086, doi:10.1016/j.eswa.2008.03.020.
[8] D. Singhal, K.S. Swarup, Electricity price forecasting using artificial neural networks, Int. J. Electr. Power Energy Syst. 33 (3) (2011) 550–555, doi:10.1016/j.ijepes.2010.12.009.
[9] S.F. Ardabili, B. Najafi, S. Shamshirband, B.M. Bidgoli, R.C. Deo, K. wing Chau, Computational intelligence approach for modeling hydrogen production: a review, Eng. Appl. Comput. Fluid Mech. 12 (1) (2018) 438–458.
[10] B.R. Chang, H.F. Tsai, Novel hybrid approach to data-packet-flow prediction for improving network traffic analysis, Appl. Soft Comput. 9 (3) (2009) 1177–1183, doi:10.1016/j.asoc.2009.03.003.
[11] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control Signals Syst. 5 (4) (1992) 455, doi:10.1007/BF02134016.
[12] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Netw. 2 (5) (1989) 359–366, doi:10.1016/0893-6080(89)90020-8.
[13] P. Zhang, Time series forecasting using a hybrid ARIMA and neural network model., Neurocomputing 50 (2003) 159–175, doi:10.1016/S0925-2312(01)00702-0.
[14] W.R. Foster, F. Collopy, L.H. Ungar, Neural network forecasting of short, noisy time series, Computers & Chemical Engineering 16 (4) (1992) 293–297, doi:10.1016/0098-1354(92)80049-F.
[15] M.C. Brace, J. Schmidt, M. Hadlin, Comparison of the forecasting accuracy of neural networks with other established techniques, in: Proceedings of the First International Forum on Applications of Neural Networks to Power Systems, 1991, pp. 31–35, doi:10.1109/ANN.1991.213493.
[16] S. Aras, Ä.D. Kocako, A new model selection strategy in time series forecasting with artificial neural networks: IHTS, Neurocomputing 174 (2016) 974–987, doi:10.1016/j.neucom.2015.10.036.
[17] J.W. Denton, How good are neural networks for causal forecasting, The Journal of Business Forecasting Methods and Systems 14 (2) (1995) 17–20.
[18] T.H. Hann, E. Steurer, Much ado about nothing? exchange rate forecasting: neural networks vs. linear models using monthly and weekly data, Neurocomputing 10 (4) (1996) 323–339, doi:10.1016/0925-2312(95)00137-9.
[19] J.L. Callen, C.C.Y. Kwan, P.C.Y. Yip, Y. Yuan, Neural network forecasting of quarterly accounting earnings, International Journal of Forecasting 12 (4) (1996) 475–482, doi:10.1016/S0169-2070(96)00706-6.
[20] R. Adhikari, R. K. Agrawal, A combination of artificial neural network and random walk models for financial time series forecasting, Neural Computing and Applications Accepted (2013), doi:10.1007/s00521-013-1386-y.
[21] D. Omer Faruk, A hybrid neural network and ARIMA model for water quality time series prediction, Engineering Applications of Artificial Intelligence 23 (4) (2010) 586–594, doi:10.1016/j.engappai.2009.09.015.
[22] M. Khashei, M. Bijari, A novel hybridization of artificial neural networks and ARIMA models for time series forecasting, Appl. Soft Comput. 11 (2) (2011) 2664–2675, doi:10.1016/j.asoc.2010.10.015.
[23] C.N. Babu, B.E. Reddy, A moving-average filter based hybrid ARIMA-ANN model for forecasting time series data, Appl. Soft Comput. 23 (2014) 27–38, doi:10.1016/j.asoc.2014.05.028.
[24] N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.-C. Yen, C.C. Tung, H.H. Liu, The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis, Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 454 (1971) (1998) 903–995, doi:10.1098/rspa.1998.0193.
[25] U.C. Buyuksahin, S. Ertekin, Time series forecasting using empirical mode decomposition and hybrid method, 26th Signal Processing and Communication Application Conference (SIU), IEEE, May 2018, Izmir Turkey (2018).
[26] J. Wang, W. Zhang, Y. Li, J. Wang, Z. Dang, Forecasting wind speed using empirical mode decomposition and elman neural network, Applied Soft Computing 23 (2014) 452–459, doi:10.1016/j.asoc.2014.06.027.
[27] M. Hibon, T. Evgeniou, To combine or not to combine: selecting among forecasts and their combinations. international journal of forecasting, International Journal of Forecasting 21 (2005) 15–24.
[28] T. Taskaya-Temizel, M.C. Casey, A comparative study of autoregressive neural network hybrids, Neural Networks: The Official Journal of the International

Neural Network Society 18 (5–6) (2005) 781–789, doi:10.1016/j.neunet.2005.06.003.
[29] Intraday EPIAS, (https://www.epias.com.tr/en/intra-day-market/introduction). 2018, (accessed 2018-05-14).
[30] P. Baldi, A.D. Long, A bayesian framework for the analysis of microarray expression data: regularized t -test and statistical inferences of gene changes, Bioinformatics (Oxford, England) 17 (6) (2001) 509–519.
[31] I. Moosa, K. Burns, The unbeatable random walk in exchange rate forecasting: reality or myth? Journal of Macroeconomics 40 (2014) 69–81, doi:10.1016/j.jmacro.2014.03.003.
[32] N.E. Huang, M.-L. Wu, W. Qu, S.R. Long, S.S.P. Shen, Applications of hilbert huang transform to non-stationary financial time series analysis, Applied Stochastic Models in Business and Industry 19 (3) (2003) 245–268, doi:10.1002/asmb.501.

Ümit Çavuş Büyükşahin is received the B.Sc. in computer engineering from the Middle East Technical University (METU), Ankara, Turkey, in 2011 and hold European Master in Distributed Computing from KTH Royal Institute of Technology in Stockholm, Sweden and UPC, Polytechnic University of Catalunya in Barcelona, Spain. He is currently continuing the Ph.D. studies at the METU. He is working as data scientist and focus on design, analysis and implementation of machine learning algorithms to solve real-world time series problems such as renewable energy production forecasting.



Dr. Şeyda Ertekin is an assistant professor in Department of Computer Engineering at the Middle East Technical University (METU). Prior to joining to METU, she worked as a research scientist at Massachusetts Institute of Technology (MIT) with appointments at the MIT Sloan School of Management and the Computer Science and Artificial Intelligence (CSAIL) Lab. Throughout her Ph.D. studies, she has also worked as a researcher in the Machine Learning Group at NEC Research Laboratories. She received her Ph.D. in Computer Science and Engineering from the Pennsylvania State University, her M.Sc in University of Louisiana at Lafayette and her B.Sc in Electrical and Electronics Engineering from METU. Her research interests mainly focus on the design, analysis and implementation of machine learning algorithms and approaches for Big Data analytics to solve real-world problems in various industrial domains such as energy & power, manufacturing and health.