Expert Systems With Applications 83 (2017) 164–176

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Random forests-based extreme learning machine ensemble for multi-regime time series prediction

Lin Lin*, Fang Wang, Xiaolong Xie, Shisheng Zhong

Machine building, School of Mechatronics Engineering, Harbin Institute of Technology, Harbin, China

ARTICLE INFO

Article history:
Received 10 January 2016
Revised 3 April 2017
Accepted 4 April 2017
Available online 5 April 2017

Keywords:
Extreme learning machine
Random forests
Ensemble learning
Multi-regime time series
Time series prediction

ABSTRACT

Accurate and timely predicting values of performance parameters are currently strongly needed for important complex equipment in engineering. In time series prediction, two problems are urgent to be solved. One problem is how to achieve the accuracy, stability and efficiency together, and the other is how to handle time series with multiple regimes. To solve these two problems, random forests-based extreme learning machine ensemble model and a novel multi-regime approach are proposed respectively, and these two approaches can be integrated to achieve better performance. First, the extreme learning machine (ELM) is used in the proposed model because of its efficiency. Then the regularized ELM and ensemble learning strategy are used to improve generalization performance and prediction accuracy. The bootstrap sampling technique is used to generate training sample sets for multiple base-level ELM models, and then the random forests (RF) model is used as the combiner to aggregate these ELM models to achieve more accurate and stable performance. Next, based on the specific properties of turbofan engine time series, a multi-regime approach is proposed to handle it. Regimes are first separated, then the proposed RF-based ELM ensemble model is used to learn models of all regimes, individually, and last, all the learned regime models are aggregated to predict performance parameter at the future timestamp. The proposed RF-based ELM ensemble model and multi-regime approaches are evaluated by using NN3 time series and NASA turbofan engine time series, and then the proposed model is applied to the exhaust gas temperature prediction of CFM engine. The results demonstrate that the proposed RF-based ELM ensemble model and multi-regime approach can be accurate, stable and efficient in predicting multi-regime time series, and it can be robust against overfitting.

1. Introduction

Since the data can be collected automatically from various and numerous sources, the global amount of information tends to grow rapidly in many fields of science. Although these data most likely improve the precision and details about the considered phenomena, they are also raising many new challenges (Miche et al., 2010). Accurate and timely predicting values of performance parameters are currently strongly needed for important complex equipment in engineering. For example, EGT (Exhaust Gas Temperature) as a performance indicator for aero engine is need to be predicted precisely, which can make faults confirmed in advance, then it will guarantee the flight safety and cut down the maintenance cost. Therefore, prediction technology based on big data in significant dynamical and intelligent system is urgent to be studied.

On the digital information era, time series prediction is widely applied to all fields of natural science (Ailliot & Monbet, 2012), economics (Chen, Lai, & Yeh, 2012; Xia & Wong, 2014) and engineering (das Chagas Moura, Zio, Lins, & Droguett, 2011). A number of time series prediction models have been widely used, e.g., artificial neural networks (ANN) (Chong, 2013), support vector regression (SVR) (das Chagas Moura et al., 2011), case-based reasoning (Liu & Chen, 2012), Takagi–Sugeno model (Xie, Lin, & Zhong, 2014), knowledge-based expert system (Ghanbari, Kazemi, Mehmanpazir, & Nakhostin, 2013), copula model (Patton, 2012; Rémillard, Papageorgiou, & Soustra, 2012), the model combining autoregressive integrated moving average and genetic programing (Lee & Tong, 2011), integration of fuzzy systems and ANN (Hadavandi, Shavandi, & Ghanbari, 2010). Their advantages and disadvantages are shown in Table 1.

In Table 1, ANN is artificial neural networks; SVR is support vector regression; CBR is case-based reasoning; TS is Takagi–Sugeno model; KBES is knowledge-based expert system; CM is copula model; ARIMAGP is the model combining autoregressive

* Corresponding author.
E-mail addresses: waiwaiyl@sina.com (L. Lin), paperused@sina.com (F. Wang), xiaolong.xie88@gmail.com (X. Xie), zhongss@hit.edu.cn (S. Zhong).

http://dx.doi.org/10.1016/j.eswa.2017.04.013
0957-4174/

Table 1
The advantages and disadvantages of a number of time series prediction models.

Model	Advantages	Disadvantages
ANN	High prediction accuracy	Time-consuming: slow learning phrase; trapped in a local minimum; subjectively in selecting the model architecture
SVR	The structural risk minimization (SRM) principle is applied to minimize an upper bound on the generalization error; achieve the global optimum	Not suitable for large training samples; difficult to solve multi-class problem
CBR	Experts' experience is included; relatively small computation cost	The acquisition of experience is difficult; not suitable for new case
TS	Approximate nonlinear systems effectively; not need domain knowledge	Parameter identification in the model is difficult
KBES	Natural knowledge is described; unified structure (IF…THEN…); acquisition and processing of knowledge is separate	Nontransparent knowledge; a lack of knowledge reuse
CM	Consider for both univariate time series processes and for multivariate time series processes	Optimal copula model is difficult to be selected
ARIMAGP	Possess both linear and nonlinear modeling abilities; more accurately than either of the models used separately; can easily be constructed in practice for either large or small data sets	Time-consuming
FSANN	Improve forecasting accuracy using minimum required input data and the least complex model	Time-consuming

integrated moving average and genetic programing; FSANN is integration of fuzzy systems and ANN.

In time series prediction, obtaining accurate predictions is not the only objective; some higher demands require the time series prediction approach to have stronger capabilities. The following two problems are known in engineering applications.

The first problem is how to achieve the accuracy, stability, and efficiency together. In engineering field, the accuracy is important, and the approach should also be stable, i.e., the accuracy of the approach can be repeated, and the predicted results in different trials should be similar, otherwise, the prediction cannot be trusted. As for the stability, another problem is that the approach should be robust against noise because noise always exists in time series from engineering field, where the data is obtained from sensors. Moreover, the efficiency is also important in some applications, particularly online predictions, where the rapid response is required, so the prediction approach should not be too time-consuming. How to achieve these three objectives at the same time is a problem urgent to be solved.

The second problem is how to handle multi-regime time series. A number of time series contain several different regimes or states, such as economic time series and sensor measurement time series obtained from operating equipment. As for economic time series, different policies or different markets may lead to different behaviors of time series, e.g., the bear and bull markets have different behaviors. As for complex equipment, different operational conditions, different loads or even different temperatures may lead to different operational regimes, which will also lead to different time series behaviors, and this phenomenon is common in engineering field. As for multi-regime time series, the conventional approaches, which try to use one global model to represent the time series behavior, may not perform well because the global model may not be able to represent the complex inner structure of the time series. Therefore, how to handle the multi-regime time series is an important problem.

Actually, some researches have been done to solve these two problems. As for the first problem, in order to improve the efficiency, a recently proposed and efficient neural network, which is extreme learning machine (ELM) (Huang, Zhu, & Siew, 2004), was used in time series prediction (Singh & Balasundaram, 2007). There is no iteration step in its training phase, so it is very fast. As for the stability, ensemble learning is a good choice (Melin, Soto, Castillo, & Soria, 2012). The ensemble learning model trains multiple models and then combines them together to generate a better model than the model achieved by any of the constituent learning models (Rokach, 2010), which are called base learners. The ensemble learning model can improve the accuracy and stability. Therefore, to improve the accuracy, stability, and efficiency together, a random forests-based ELM ensemble model is proposed. Random forests (RF) model is a popular ensemble algorithm, and it is used in the ELM ensemble model to improve the accuracy and stability.

As for the second problem, most of researches used Markov regime switching model (Hamilton, 2010; Kim & Nelson, 1999) to solve multi-regime time series. However, the time series in this paper, which is the turbofan engine time series, has some specific properties, which will be detailed later, so the Markov regime switching approach may not be suitable. Therefore, a novel multi-regime approach for this type of time series is proposed. The RF-based ELM ensemble and proposed multi-regime approach are used together in the turbofan engine time series prediction.

This paper is structured as follows. Section 2 will introduce the ELM and RF algorithm, which will be used in the proposed time series prediction model. Section 3 will detail the methodology, where Section 3.1 will present the RF-based ELM ensemble model, and Section 3.2 will show the specific properties of turbofan engine time series and detail the proposed multi-regime approach. Experiments will be executed in Section 4. Section 4.1 will use publicly available NN3 time series to validate the proposed RF-based ELM ensemble model; Section 4.2 contains three parts, the first two parts will use the simulated turbofan engine time series from NASA to evaluate the RF-based ELM ensemble model and multi-regime approach, respectively; and in the third part, the RF-based ELM ensemble model will be applied to the real time series prediction of CFM engine.

2. Related works

This section will introduce the ELM and RF models, which will be used in the proposed model.

2.1. Extreme learning machine

ELM (Huang et al., 2004) is a single-hidden layer feedforward neural networks (SLFNs), where the input weights are randomly initialized, then fixed without iteratively tuning, and the output weights are calculated analytically (Huang, Huang, Song, & You, 2015). There is no iteration in the training phase of ELM, so it is very fast. Besides, empirical studies also showed that the generalization ability of ELM is comparable or even better than that of support vector machine and its variants (Huang et al., 2015). The structure of SLFNs is shown in Fig. 1.

There is only one hidden layer in SLFNs. Suppose the training sample set is $Train = \{(\mathbf{X}^i, \mathbf{Y}^i) | \mathbf{X}^i \in \Re^m, \mathbf{Y}^i \in \Re^n\}, i = 1, 2, \ldots, N$,

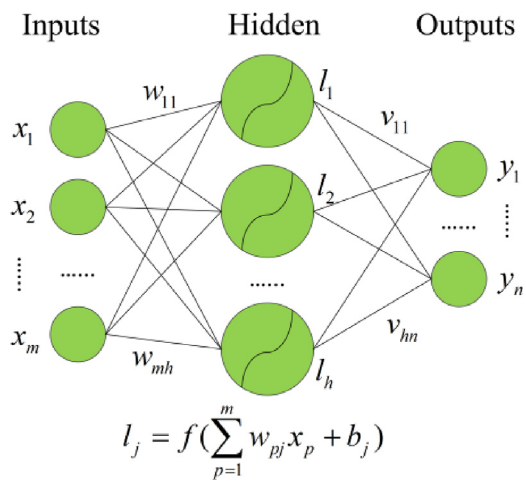


Fig. 1. The structure of SLFNs.

where N is the number of training samples, so the SLFNs have m inputs and n outputs, i.e., $\mathbf{X}^i = [x_1^i, x_2^i, \ldots, x_m^i]$ and $\mathbf{Y}^i = [y_1^i, y_2^i, \ldots, y_n^i]$. In order to be simple, the output number n is set 1 in the following contents, and the ELM used in this paper also has only one output. Suppose the SLFNs have h hidden neurons, so the output of the SLFNs can be calculated as $o^i = \sum_{j=1}^{h} v_j \cdot f(\sum_{p=1}^{m} w_{pj} \cdot x_p^i + b_j), i = 1, 2, \ldots, N$, where w_{pj} represents the input weight from the pth input node to the jth hidden neuron, v_j is the output weight from the jth hidden neuron to the output node, b_j is the bias of the jth hidden neuron, and $f(\cdot)$ is the activation function in hidden neurons. In ELM, all the input weights $w_{pj}, p = 1, \ldots, m; j = 1, \ldots, h$, and all the biases $b_j, j = 1, \ldots, h$ are randomly initialized, and their values remain unchanged during the training phase, so only the output weights $v_j, j = 1, \ldots, h$ need to be calculated. Suppose that

$$\mathbf{L} = \begin{bmatrix} f\left(\sum_{p=1}^{m} w_{p1} \cdot x_p^1 + b_1\right) & f\left(\sum_{p=1}^{m} w_{p2} \cdot x_p^1 + b_2\right) & \cdots & f\left(\sum_{p=1}^{m} w_{ph} \cdot x_p^1 + b_h\right) \\ f\left(\sum_{p=1}^{m} w_{p1} \cdot x_p^2 + b_1\right) & f\left(\sum_{p=1}^{m} w_{p2} \cdot x_p^2 + b_2\right) & \cdots & f\left(\sum_{p=1}^{m} w_{ph} \cdot x_p^2 + b_h\right) \\ \vdots & \vdots & \cdots & \vdots \\ f\left(\sum_{p=1}^{m} w_{p1} \cdot x_p^N + b_1\right) & f\left(\sum_{p=1}^{m} w_{p2} \cdot x_p^N + b_2\right) & \cdots & f\left(\sum_{p=1}^{m} w_{ph} \cdot x_p^N + b_h\right) \end{bmatrix}_{N \times h},$$

$$\theta = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_h \end{bmatrix}_{h \times 1}, \quad \text{and} \quad \mathbf{T} = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^N \end{bmatrix}_{N \times 1},$$

and then the ELM model outputs can be represented by $\mathbf{O} = \mathbf{L} \cdot \theta$, where $\boldsymbol{\theta}$ is the undetermined output weights vector. The aim is that the model outputs $\mathbf{O}$ equal the actual outputs $\mathbf{T}$, so if the following equation holds,

$$\mathbf{T} = \mathbf{L} \cdot \theta, \tag{1}$$

then the output weights can be calculated as the following equation

$$\theta = \mathbf{L}^{\dagger} \cdot \mathbf{T}, \tag{2}$$

where $\mathbf{L}\dagger$ is the Moore–Penrose generalized inverse of matrix $\mathbf{L}$. Then the ELM model can be obtained; thus, there is no iteration in training ELM model, only a pseudo-inverse needs to be calculated, which is very efficient. Because of its efficiency and good generalization performance, ELM is widely studied, and previous researches demonstrated that it can perform well in regression, classification (Huang, Zhou, Ding, & Zhang, 2012), time series prediction (Singh & Balasundaram, 2007), etc. Most researches focused on improving its capability because ELM is a new technique, as for the engineering applications, some can be found, e.g., anomaly detection in traffic (Wang, Li, Du, & Pan, 2015), medical diagnosis application (Huang, Zhu, & Siew, 2006).

2.2. Random forests

The RF model (Breiman, 2001) is a widely used ensemble learning model. Compared with the global machine learning models, such as ANN, SVR, which try to build one global model from the data, the ensemble learning model, which tries to build a set of models and then combines them together, may perform better, particularly when dealing with complex systems. Generally, the ensemble learning model will perform better than any base learners if the adequate ensemble strategy is adopted. There are several widely used ensemble strategies, such as Bagging (Breiman, 1996), Boosting (Freund & Schapire, 1997), and random subspace technique (Ho, 1998). All these strategies are used to achieve the diversity of base learners, which is very important for ensemble learning model to achieve a good result. The diversity means that all the base learners in an ensemble are different from each other.

As an ensemble learning model, RF has its own base learners and ensemble strategies, where the base learners are all classification and regression trees (CARTs), and RF model uses two strategies, which are Bagging and random subspace technique. The RF algorithm is illustrated in Fig. 2.

There are three steps in RF algorithm, where Step 1 uses the sampling with replacement technique, which is also known as bootstrap sampling technique, to generate n different training sample sets for n CARTs because of its stochastic property. Then the random subspace technique is used in Step 2. As for each node of each CART, partial features are selected from all input features randomly. As is shown in Fig. 2, every sample has m input features in Step 1, while in Step 2, every node of every CART only uses p input features, where $p < m$. Therefore, all n CARTs have different training sample sets and all nodes have different features, so after

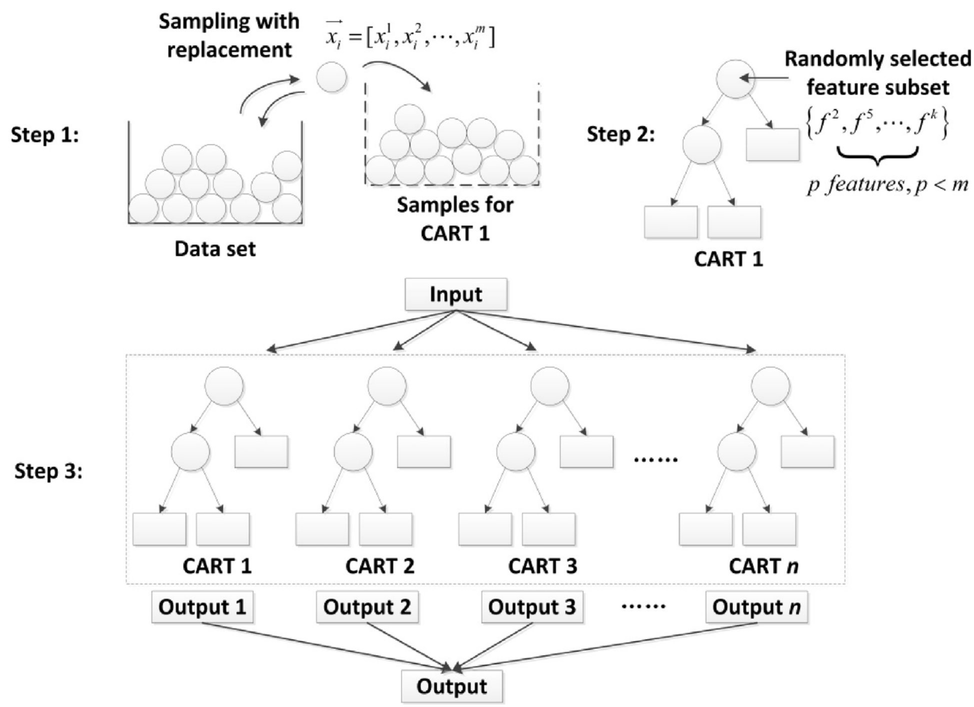


Fig. 2. Random forests algorithm.

training phase, n different CARTs can be obtained, and the diversity is achieved. Last, in Step 3, n different CARTs are aggregated to generate one model. In the test phase, the test inputs are fed into n different CARTs simultaneously, and then the output of the ensemble model is calculated by aggregating the n outputs of n CARTs, which is majority voting for classification problem and averaging for regression problem. The algorithm for regression is detailed as follows (Liaw & Wiener, 2002):

Step 1: Generate n bootstrap sample sets from the original dataset.
Step 2: For each bootstrap sample set, grow an unpruned regression tree with the following modification: at each node, randomly sample p features from all the input features and choose the best split from those p features, where $p< m$, and m is the number of all input features.
Step 3: Predict new output by averaging the outputs of n regression trees when new inputs are fed into RF.

As is shown in the algorithm, only two parameters, which are number of trees n and feature number of each node p, are required to be determined by users. Moreover, it has been demonstrated that RF model is generally not sensitive to the values of these two parameters. The users do not need to spend too much time on tuning the parameters when the model is used on a new dataset. The users only need to select the parameters according to the recommendation, or even select them arbitrarily, and then results of RF do not vary greatly with parameters (Liaw & Wiener, 2002). Moreover, random forests do not overfit as more independent and diversiform trees are added (Breiman, 2001), which makes the application of RF model even more convenient.

3. Methodology

The methodology contains two parts: the RF-based ELM ensemble model and the approach that can handle the turbofan engine multi-regime time series, which has its specific properties. The details will be presented as follows.

3.1. Random forests based extreme learning machine ensemble

As shown in Section 2.1, only a pseudo-inverse needs to be calculated in ELM, so it is faster than ANN and SVR where there are a number of iterations in the training phase .while just as other machine learning models, it tries to minimize the training error of the entire training dataset, which may lead to overfitting because ELM will try to approximate all training samples well (Liu & Wang, 2010). The risk of overfitting can be reduced by two approaches, the first one is the regularization, and the second one is using the ensemble learning strategy (Huang et al., 2015). These two approaches are both adopted in this paper.

A good generalization performance makes the ELM accurate. The concept of regularization – using L_1, L_2 or other norms-based penalties on the output weights has been studied extensively (Miche et al., 2010; Miche, van Heeswijk, Bas, Simula, & Lendasse, 2011). Originally, according to Bartlett's theory (Bartlett, 1998), for feedforward neural networks reaching smaller training error, the smaller the norms of the weights are, the better generalization performance of the networks tend to have. Based on Bartlett's theory, the regularized ELM model was proposed (Deng, Zheng, & Chen, 2009), and it tried to minimize the training error as well as the norm of the output weights, which is,

$$Minimize : E = \frac{1}{2}\left\|\boldsymbol{\theta}\right\|_2^2 + C\cdot\frac{1}{2}\sum_{i=1}^{N}\left|\boldsymbol{e}^i\right|^2 = \frac{1}{2}\cdot\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\theta} + \frac{C}{2}\cdot\left(\mathbf{L}\cdot\boldsymbol{\theta}-\mathbf{T}\right)^{\mathrm{T}}\cdot\left(\mathbf{L}\cdot\boldsymbol{\theta}-\mathbf{T}\right), \tag{3}$$

where $\frac{1}{2}\|\boldsymbol{\theta}\|_2^2$ is the regularization term, which is used to avoid overfitting and improve the generalization performance of the ELM model, and C is a constant. Actually, this is a ridge regression problem if the 2-norm $\|\cdot\|_2$ is used in the regularization term. The reason why 2-norm is selected is because there will be an analytical solution, otherwise, no analytical solution exists and iteration algorithm should be used to find the minimum, then the running time may be longer. The analytical solution of the objective function in Eq. (3) is

$$\boldsymbol{\theta} = \left(\mathbf{L}^{\mathrm{T}}\mathbf{L} + \frac{\mathbf{I}}{C}\right)^{-1} \cdot \mathbf{L}^{\mathrm{T}}\mathbf{T}. \tag{4}$$

The above contents detail the regularized ELM, and the following contents will present the ELM ensemble model. The ELM ensemble, whose base learners are ELM models, is efficient and stable. The key point is the ensemble strategy or the aggregation approach. Several ELM ensembles have been designed, e.g., voting based ELM ensemble for classification problem (Cao, Lin, Huang, & Liu, 2012), ensemble of online sequential ELM (Lan, Soh, & Huang, 2009), ELM ensemble using cross validation (Liu & Wang, 2010). All these ELM ensemble models used simple aggregation strategies to combine all the base-level ELM models, which are majority voting for classification and averaging for regression. In this paper, a RF based ELM ensemble model is proposed to improve the accuracy and stability of the ensemble model.

As mentioned above, one important thing for ensemble is the diversity of base learners. Actually, as for every ELM, its input weights and biases of hidden neurons are all randomly assigned, so all the ELMs in one ensemble are already quite different. In order to achieve larger diversity, the widely used bootstrap sampling technique, which is also used in RF model, is adopted to generate the training samples for every ELM. Similarly, after using this technique, different training sample sets are obtained for different ELM models, and diverse ELM models can be learned.

The second problem is how to aggregate the outputs of all ELMs to obtain the final output of the ensemble. Bagging strategy uses the average, and this simple strategy can achieve very good results. Some other aggregation approaches were proposed and used, such as the weighting approach, dynamic weighting approach, selective weighting approach. The above aggregation approaches can be seen as statistical aggregations, and some researchers tried to use machine learning algorithms in aggregation because of their good non-linear approximation ability. Gheyas and Smith (2011) used neural networks as the combiner to assign the weights of base learners. They proposed a novel homogeneous neural network, which was called generalized regression neural network (GRNN), and then in the ensemble model, the base learners were GRNNs, and the outputs of these base-level GRNNs were combined using a combiner GRNN to obtain the final output. Inspired by their researches, the machine learning algorithm is used as the combiner in this paper, while neural networks have some drawbacks. The most important one is its stability. The neural networks are sensitive to the initial values and not stable. Moreover, the neural networks may suffer overfitting. Massive researches have been done to overcome these drawbacks when using neural networks, while these techniques may increase the complexity of the model. Thus, RF model is used as the combiner in the ELM ensemble because it is a more stable and more accurate model than neural networks, and it is robust against overfitting. Breiman (2001) proved that in random forests which is an ensemble model, as the number of trees increases, for almost surely all sequences $\Theta_1, \ldots, \Theta_k$, the generalization error PE^* converges to

$$P_{X,Y}\left(P_{\Theta}(h(X,\Theta) = Y) - \max_{j \neq Y} P_{\Theta}(h(X,\Theta) = j) < 0\right) \tag{5}$$

where, X, Y, Θ-random vector;

k-number of trees;

$h(X, \Theta)$- classifier.

Considering the above statement, the proposed model can effectively avoid overfitting because RF model is used.

In its crudest form, time series prediction problem is similar to regression problems. The RF-based ELM ensemble model is a generic regression model, so it can be applied to solve different problems. In time series prediction, several previous timestamps are used to predict performance parameter at the future timestamps. Formally, suppose that in the time series $\{x_t\}, t = 1, \ldots, N$, m previous timestamps are used to predict the future one timestamp, i.e., $(x_i, x_{i+1}, \ldots, x_{i+m-1})$ are used to predict x_{i+m}, where $i = 1, 2, \ldots, N - m$. Therefore, every adjacent $m + 1$ timestamps are used to generate one sample, and a time series that contains N timestamps can generate $N - m$ samples, and every sample contains m inputs and one output. Then all the training samples are used to train the RF-based ELM ensemble model. The training phase contains two steps, which are shown in Fig. 3.

Step 1: Train M ELM models individually by using M bootstrap sample sets (Fig. 3(a)).

Step 2: Train the RF combiner model by using all the samples and the trained ELM models. As is shown in Fig. 3(b), the inputs of one sample $(x_i, x_{i+1}, \ldots, x_{i+m-1})$ are fed into all M ELM models, and M different output estimates, which are $x^l_{i+m}, l = 1, \ldots, M$, can be obtained. Then, all these output estimates are used as the inputs of RF model, and the actual output x_{i+m} is used as the output of the RF model to train the RF model, which has M inputs and one output. After training, the trained RF-based ELM ensemble model can be used to predict the future timestamp value.

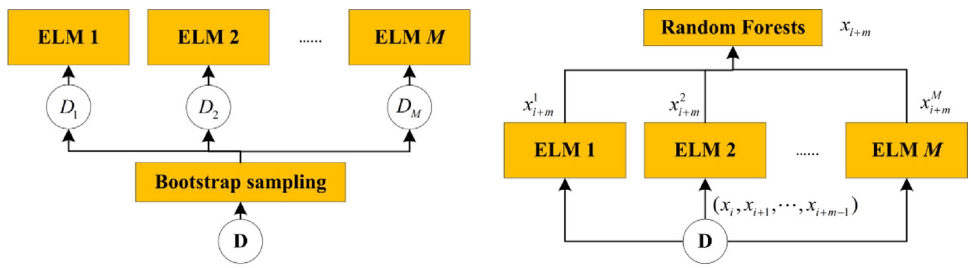
In a word, random forests based extreme learning machine ensemble model is faster than ANN and SVR that need iterations in the training phase. The generalization ability of the proposed model can be guaranteed with regularization and ensemble strategy, although SVR and ANN also have their own regularization methods and RF has its own ensemble strategy. But the proposed model synthesizes both methodologies to avoid overfitting. The drawback of the proposed model is that it only can predict performance parameter at the next timestamp.

3.2. Multi-regime approach

As mentioned in Section 1, multi-regime time series is common in engineering field, and most of previous researches use Markov regime switching model to handle it. This model supposes that there are a series of hidden states (regimes) in the time series, and different states have different processes. The transition of states is stochastic, but the dynamics behind the switching process is known and driven by a transition matrix, which means that the regime-switching process has Markov property, where the future state depends only upon the present state, not on the sequence of previous states.

However, the multi-regime time series from turbofan engine has its specific properties. The regimes in turbofan engine time series are already known, which are determined by the operational parameters, while in the Markov switching model, the assumption is that the regimes are unknown. Moreover, there is no time duration for each operational regime in the turbofan engine time series. The Markov switching model may not be suitable for this time series because of these two specific properties, and the details are shown as follows.

The multi-regime turbofan engine time series is obtained from the turbofan engine degradation simulation data that is publicly available from NASA Ames Prognostics Data Repository (Saxena, Goebel, Simon, & Eklund, 2008), which is designed to estimate the remaining useful life of turbofan engine originally. The simulation model was built on Commercial Modular Aero-Propulsion System Simulation (CMAPSS) developed in NASA Army Research Laboratory (Frederick, DeCastro, & Litt, 2007). The collected data for each engine provides a track of the engine's condition throughout its usage history in the format of a 24-dimentional time series, where the first three parameters denote operational conditions and the remaining 21 parameters are sensor measurements at each flight



a) Step 1: base-level ELM training b) Step 2: RF combiner training

Fig. 3. The training phase of RF-based ELM ensemble.

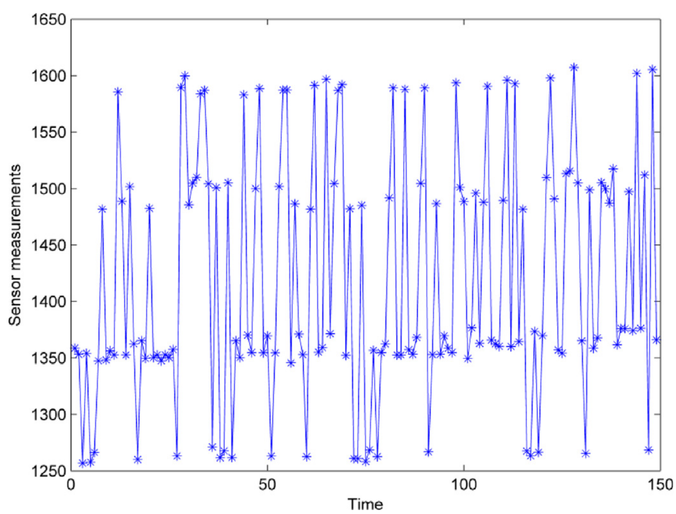


Fig. 4. The time series of one sensor measurement of one turbofan engine.

cycle, where one flight cycle is one take-off and one landing. As for each flight cycle, only one snapshot of 24 parameters is taken to represent the engine condition of the current flight cycle. Therefore, in this turbofan engine time series, every timestamp represents one flight cycle. Only the univariate time series is considered in this paper, thus, 21 sensor measurements generate 21 time series. The time series of one engine's one sensor measurement is shown in Fig. 4.

The operational regime, which is determined by the first three operational parameters, is known to users. These three operational parameters are altitude, Mach number (speed) and Throttle Resolver Angle (TRA) value (thrust setting) (Wang, 2010). These three parameters setting of the following flight cycle may have no relationship with the parameter setting in the previous flight cycle, i.e., the operational regime in the following flight cycle may not depend on the previous flight cycle, so maybe there is no relationship between the regimes of two adjacent flight cycles. Therefore, there may be no Markov property in the regime-switching process in this time series. The regime-switching process of the time series in Fig. 4 is shown in the following figure.

There are six operational regimes totally, and the star in the figure represents the regime of every flight cycle, and it can be observed that there is no time duration for every operational regime, and regimes can switch frequently. Moreover, the regimes of historical time series are all known, so the Markov regime switching approach may not be suitable. Therefore, a new approach is proposed to handle this type of time series.

Suppose the regimes do not have the Markov property, they only depend on the probabilities of operational regimes in each flight cycle. Suppose T regimes exist, and their probabilities in one flight cycle are $p_1, p_2, \ldots, p_T$, where $\sum_{t=1}^{T} p_t = 1$. Every flight cycle has p_t probability to be in the tth operational regime, and every operational regime has its own time series behavior, thus, the proposed RF-based ELM ensemble is used to model the time series of every operational regime individually, and then different regime models are aggregated.

As for the RF-based ELM ensemble time series model, m previous timestamps $(x_i, x_{i+1}, \ldots, x_{i+m-1})$are used to predict the future one timestamp x_{i+m}, while in one sample $(x_i, x_{i+1}, \ldots, x_{i+m-1}; x_{i+m})$, all of these $m+1$ timestamps may belong to different operational regimes, so the problem is how to classify the training samples into different operational regimes. In this approach, the operational regime of every sample depends on the regime of the output timestamp, i.e., $R(x_i, x_{i+1}, \ldots, x_{i+m-1}; x_{i+m}) = R(x_{i+m}), i = 1, 2, \ldots, N-m$, where $R(\cdot)$ returns the operational regime, because it supposes that the operational regimes of adjacent timestamps have no relationship. The multi-regime approach and RF-based ELM ensemble model are integrated, and the approach is detailed as follows.

Step 1: Regime separation.
First, the regime of every timestamp is obtained based on the operational parameters. Then, the time series is transformed into training samples. Every adjacent $m+1$ timestamps are used to generate one sample and $N-m$ samples are obtained. Next, the regime of every sample is determined by the regime of its $(m+1)$-th timestamp. Last, the samples in the same regimes are classified into one set, so T different sample sets can be obtained. Every set contains all the samples that can represent the behavior of one operational regime.

Step 2: Regime models training.
After dividing the samples into different regimes, the next step is training different regime models individually by using these divided samples. Therefore, T different models can be obtained and every model represents the behavior of one operational regime. As for the modeling algorithm, it can be any regression model, e.g., ANN, SVR, ELM, etc. The proposed RF-based ELM ensemble model is used to model the time series in every regime.

Step 3: Multiple regime models aggregation.
The first two steps can be seen as training phase, and then the obtained T models are used to predict the future timestamp value. The problem is that the regime of future timestamp is unknown. Because the regimes of adjacent flight cycles are assumed have no relationship. That is the regime in every circle for planes is completely independent and probabilities p_i for the regimes are unpredictable. As mentioned above, the probabilities of the future flight cycle in T regimes are $p_1, p_2, \ldots, p_T$, where $\sum_{t=1}^{T} p_t = 1$, and

based on the obtained operational regime models, the prediction of future timestamp value is obtained using the following equation, which is

$$x_j = \sum_{t=1}^{T} p_t \cdot M_t(x_{j-m}, \dots, x_{j-1}), \tag{6}$$

where $M_t(\cdot)$ represent the model of the tth regime. Then the problem is how to estimate the probabilities $p_1, p_2, \dots, p_T$. Two solutions are used in this paper. In the first solution, the probabilities of all regimes are equal, i.e., $p_1 = p_2 = \dots = p_T = \frac{1}{T}$, and it is called equiprobability solution. In the second solution, the frequencies of different regimes in the training samples are used as the estimates of their probabilities, so it is called frequency-based solution. Suppose the number of training samples in T regimes are $N_1, N_2, \dots, N_T$, where $N_1 + N_2 + \dots + N_T = N - m$, so the probabilities can be calculated as

$$p_t = \frac{N_t}{N-m}, t = 1, 2, \dots, T, \tag{7}$$

and then the future timestamp can be predicted using Eq. (6) when its regime is unknown.

4. Experiments and application

In this section, the proposed model is evaluated by using a benchmark time series and then applied to the turbofan engine time series prediction.

4.1. Experiments using NN3 time series

The proposed model is evaluated by using NN3 time series (http://www.neural-forecasting-competition.com/NN3), which are drawn from the homogeneous population of empirical business time series. A total of 111 time series exist in the NN3 dataset, and the second one (NN3_002) is used in the experiments. As for the multi-regime approach, it will be evaluated in next subsection because there is no regime-switching in NN3 time series. This time series contains 69 points, where the first 40 points are used as training points, and the left 29 ones are used as test points, which are shown in Fig. 6.

It can be observed that great fluctuations exist in time series, which will make it difficult to achieve good prediction accuracy, and that is the reason why it is used to evaluate the proposed model. Besides the proposed model, some other widely used models are also executed to do the comparison, which are artificial neural network with back propagation algorithm (BPNN), SVR, ELM. Some other ensemble models are also included in the experiments, which are ELM ensemble using average aggregation approach (ELM-Avg), BPNN-Avg, the RF-based BPNN ensemble (BPNN-RF), SVR-Avg, and SVR-RF, where the first term represents the base learner and the second term represents the aggregation approach, and the proposed RF-based ELM ensemble model is denoted by ELM-RF. The SVR model is implemented by using LIBSVM toolbox (Chang & Lin, 2011).

Two error measurements are used to evaluate the accuracy of models, which are average relative error (ARE), $\text{ARE} = \frac{1}{N}\sum_{i=1}^{N}\text{RE}_i = \frac{1}{N}\sum_{i=1}^{N}\frac{|x_i - o_i|}{|x_i|}$, and root mean squared error (RMSE), $\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - o_i)^2}$, where x_i and o_i are the actual and predicted outputs of the ith test sample, and N is the number of test samples. In order to evaluate the stability of models, every model is executed 10 trials, the mean values of two error measurements are used to reflect the accuracy, and the variances of these error measurements are used to represent the stability. The efficiency of models is also evaluated, so the total running time of every model is recorded and compared. The CPU is an Intel Core i7 (2.4 GHz) with 16GB RAM. In the experiments, the numbers of hidden neurons of BPNN and ELM are all set 20, and the numbers of base learners in all ensemble models are set 20. The activation function of BPNN model is set as hyperbolic tangent sigmoid transfer function, and the activation function of ELM model is sigmoid function. As for SVR, the ε-SVR is adopted and the RBF kernel function is used.

The previous 4 timestamps are used to predict the fifth timestamp, i.e., $m = 4$. Therefore, 40 training points generate 36 training samples, and 25 test samples can be obtained from 29 test points. The experiment results of all models are presented in Table 2, where one row represents the results of one model, and one column shows the results of different models. Every model is executed 10 trials, so the first two columns are the mean values of ARE and RMSE, respectively, and the third and fourth columns denote the variances of these two error measurements. The fifth column is the total running time of 10 trials. The best result of each column is denoted by italic and boldface fonts. The 10 trial results are also illustrated as boxplots in Fig. 7, which use the horizontal format, and the width of boxes can represent the stability of the model.

On one hand, as for the accuracy, the proposed model (ELM-RF) achieves the minimal ARE, and the BPNN-RF achieves the best RMSE, where the RMSE of ELM-RF is the second best one, thus, the results show that the accuracy of the proposed model is very good, and the proposed RF-base ensemble approach is effective. Then, by comparing the error measurements of ensemble models and base learners, it can conclude that the ensemble model performs better than base learners, e.g., BPNN-RF and BPNN-Avg outperform BPNN, which is clear in Fig. 7. The exception is SVR model, SVR and SVR-Avg has the same error, and the reason will be given later. Next, as for two different aggregation approaches, the RF approach is better than averaging approach generally, e.g., ELM-RF is better than ELM-Avg and BPNN-RF is better than BPNN-Avg.

On the other hand, as for the stability, the best models are SVR and SVR-Avg models, which are exactly the same, and this also means that the ensemble model is not useful if the base learner is SVR. The reason is that the predicted results of SVR are almost the same in different trials, which means that there is no diversity among different SVR models, so the ensemble cannot improve the accuracy of SVR, and the error measurements of SVR and SVR-Avg are exactly the same. Then, by comparing the stabilities of base learner and ensemble model, e.g., BPNN and BPNN-RF or BPNN-Avg, we can see that the stability can be improved by using the ensemble model, particularly for BPNN model. BPNN is very unstable, while the BPNN ensemble models are very stable.

As for the running time, the most efficient model is SVR model. The LIBSVM toolbox is very efficient. By comparing three base learner models, BPNN is very time-consuming, and ELM is also very efficient. Of course, the ensemble models will cost more time than base learners, but the ELM ensembles and SVR ensembles are also very efficient.

By considering the accuracy, stability and efficiency together, it can conclude that the proposed RF-based ELM ensemble model is a good choice. Its accuracy is very good, and its stability and efficiency are both the second best, which are not as good as SVR model, but the accuracy of SVR models is not very good. Therefore, the proposed model outperforms other models if three aspects are considered together.

The above experiments only evaluate the performance of the proposed RF-based ELM ensemble model, and the multi-regime approach will be evaluated in next section.

Table 2
The results of different models on NN3 time series.

	ARE	RMSE	Variance (ARE)	Variance (RMSE)	Time (s)
BPNN	0.1045	591.8325	0.0024	4.8382e+4	3.8337
SVR	0.0586	403.3076	***0***	***3.5902e−27***	***0.0135***
ELM	0.0561	387.1779	7.9992e−6	349.6872	0.0552
ELM-RF	***0.0508***	360.1643	6.0819e−6	243.0934	3.7709
ELM-Avg	0.0536	371.6102	1.4575e−6	13.1364	1.4250
BPNN-RF	0.0559	***355.0662***	3.3827e−5	322.0198	111.1591
BPNN-Avg	0.0577	361.7542	1.8153e−5	673.3405	111.9993
SVR-RF	0.0767	418.0443	4.2784e−6	58.8873	2.3498
SVR-Avg	0.0586	403.3076	***0***	***3.5902e−27***	0.5157

Table 3
The results of different models on FD001 data.

	ARE	RMSE	Variance (ARE)	Variance (RMSE)	Time (s)
BPNN	0.0044	8.6021	5.1072e−7	1.6978	2.6195
SVR	0.0036	7.1735	***2.0898e−37***	***3.5060e−30***	***0.0234***
ELM	0.0042	8.6476	7.6424e−8	1.3388	0.0606
ELM-RF	***0.0034***	***6.8946***	4.2729e−10	0.0026	3.0507
ELM-Avg	0.0041	8.2967	2.9121e−9	0.0099	1.9844
BPNN-RF	0.0036	7.3343	3.7983e−8	0.1795	144.3733
BPNN-Avg	0.0035	7.2095	2.2122e−8	0.0823	157.4082
SVR-RF	0.0036	7.3648	1.0525e−8	0.0500	2.6809
SVR-Avg	0.0036	7.1735	***2.0898e−37***	***3.5060e−30***	0.7594

4.2. Application to turbofan engine time series

The RF-based ELM ensemble model is accurate, stable and efficient, so it is applied to turbofan engine time series prediction in this section. As mentioned above, the simulated dataset from NASA repository is used to evaluate the performance of the proposed model (Section 4.2.1) and the proposed multi-regime approach (Section 4.2.2), respectively, and the proposed model is applied to predicting the real exhaust gas temperature (EGT) time series from CFM engine (Section 4.2.3).

NASA repository provides 4 datasets, and different datasets have different fault modes and operational regime numbers. Datasets 1 and 2 only have one fault mode, while datasets 3 and 4 have two. As for the number of operational regimes, datasets 1 and 3 only include one, and datasets 2 and 4 have 6 operational regimes. In each dataset, a number of engines' degradation time series are included. Every dataset contains training set and test set. In training set, all the engines are run to failure, while in the test set, only partial time series are recorded. Therefore, the time series in training set is used in the following experiments. Because the fault mode is not considered in this paper, only datasets 1 and 2 are used. Section 4.2.1 will use dataset 1, which is also called FD001, to evaluate the proposed RF-based ELM ensemble model because there is only one operational regime, and then dataset 2, which is FD002 dataset, is used to evaluate the proposed multi-regime approach in Section 4.2.2.

4.2.1. Experiments using FD001 dataset

There are 100 engines in FD001 dataset, and every engine has 21 sensor measurements. The third sensor time series of the first engine is used to do the experiments, which is shown in Fig. 8.

There is only one operational regime in FD001 dataset, so the experiments are similar to those in Section 4.1. There are 192 timestamps in this time series, and in the experiments, the first 40 timestamps are used as training points, and the left 152 points are used to evaluate the model. The previous 4 timestamp values are used to predict the fifth timestamp; thus, 36 training samples and 148 test samples are obtained. The same models and the same error measurements are used. Similarly, all BPNN and ELM models have 20 hidden neurons, and all ensemble models have 20 base learners. The results are shown in Table 3, where the best one of every column will be denoted by italic and boldface font. The boxplots of these experiments are also shown in Fig. 9.

Similar conclusions can be obtained. On one hand, the proposed model outperforms other models in accuracy. It achieves the minimum in both ARE and RMSE. By comparing the base learner model and ensemble model, it can be observed that the ensemble model can improve the accuracy and stability generally, except SVR model, and the reason is given in Section 4.1.

On the other hand, the stability of the proposed model is better than others except SVR related models. The prediction errors of SVR related models are deterministic. The reason is that the algorithm in LIBSVM toolbox is optimized and there is no diversity among different SVR models.

SVR is also the most efficient one, and ELM is much better than BPNN in efficiency. As for the ensemble models, the proposed model is also very efficient. Therefore, the proposed model performs well in accuracy, stability and efficiency when it is applied to turbofan engine time series prediction.

4.2.2. Experiments using FD002 dataset

In this section, the time series from FD002 dataset is used to evaluate the proposed model and the proposed multi-regime approach because 6 operational regimes exist in this dataset. There are three steps in the multi-regime approach, and the details are presented as follows.

Step 1 is regime separation. First, the regime of every flight cycle is identified. As mentioned above, the operational regime is only determined by three operational parameters; thus, the regime identification approach, which is used by Wang (2010), is adopted, i.e., three variables for operational conditions are used to cluster the operational regimes. The clustering algorithm is used to cluster all three operational parameters of all flight cycles of all 260 engines, and then 6 cluster centers can be obtained, which are $\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_6, \mathbf{V}_t = [Alt_t, Mach_t, TRA_t]$, where *Alt, Mach* and *TRA* represent three operational parameters. As for one flight cycle *C*, its three operational parameters $[Alt^C, Mach^C, TRA^C]$ are used to classify this flight cycle into one operational regime, i.e., $R(C) = R^C = \min_{t=1,\cdots,6}(\|[Alt^C, Mach^C, TRA^C] - V_t\|)$, and the regime of every flight cycle is identified. Then, one sensor of one engine is selected and its run-to-failure records are used as a time series. In the follow-

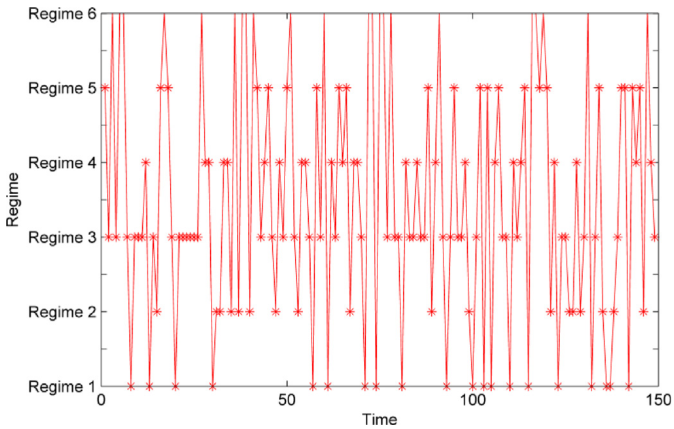


Fig. 5. The regime-switching process of turbofan engine time series.

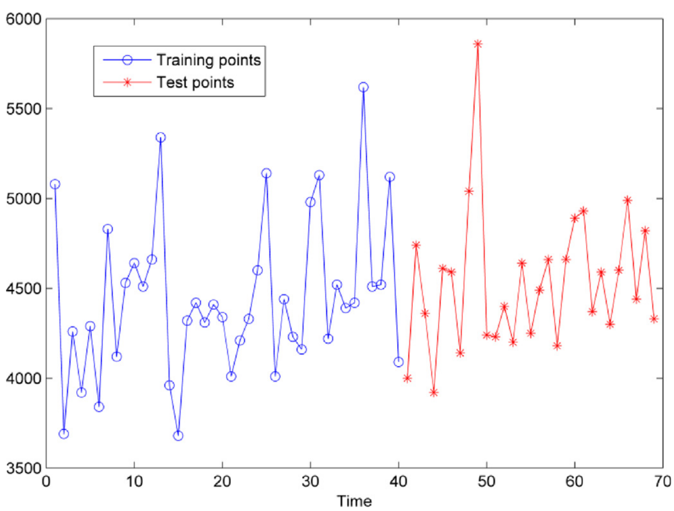


Fig. 6. The training and test points of NN3 time series.

ing experiments, the third sensor measurement of the first turbofan engine is selected, the time series is shown in Fig. 4, and the regimes of this time series are shown in Fig. 5.

Next, the time series is transformed into training samples. In the following experiments, the previous 4 timestamps are used to predict the fifth timestamp. The first 104 points are used as training points and the left 45 points are used as test points; thus, 100 training samples and 41 test samples are obtained. The operational regime of every training sample is determined by the regime of its output timestamp, i.e., the fifth timestamp in every training sample. Last, the training samples that belong to the same operational regime are classified into one sample set; thus, 6 sample sets are obtained.

Step 2 is training different regime models by using different training sample sets, respectively. In this step, different algorithms are used to train the models by using the obtained 6 training sample sets to do the comparison. All the algorithms can be classified into two categories. In the first category, the algorithms do not use the multi-regime approach. In this category, three base learner models, which are BPNN, SVR and ELM, two ensemble models, which are ELM-RF and ELM-Avg, are used to train the model directly without using the multi-regime approach to preprocess the time series (Step 1), and they train the model use the 100 training samples directly. While the models in the second category all use the multi-regime approach to predict the time series, and only the algorithms that are used to train 6 regime models are different, e.g., as is shown in the following Tables 4 and 5, in "M-BPNN", the term "M" denotes the model uses the multi-regime approach, and "BPNN" means that this model uses BPNN to train 6 regime models; "M-ELM-RF" uses the multi-regime approach, and the RF-based ELM ensemble model is used to train regime models.

In Steps 3 and 6 regime models are used to predict the future flight cycle, respectively, and then these 6 predicted values are aggregated based on the probabilities of 6 operational regimes. Two solutions, which are equiprobability solution and frequency-based solution, are used, and the experiment results of these two solutions are shown in Tables 4 and 5. The error measurements are the same with those in previous sections. The boxplots of the first solution are shown in Fig. 10.

First, as for the accuracy, by comparing the models that use multi-regime approach and models that do not use multi-regime approach, e.g., BPNN and M-BPNN, ELM and M-ELM, we can see that the models using multi-regime approach are more accurate than models that do not use multi-regime approach, which is also clearly illustrated in Fig. 10. Therefore, the proposed multi-regime approach is effective in handling multi-regime time series. Then, as for the proposed model that uses multi-regime approach (M-ELM-RF), its accuracy is very good. It achieves the minimum in RMSE when using two solutions. As for ARE, although it is not the best one, its ARE is very near the minimum, so it demonstrates that the proposed approach is accurate. Moreover, by comparing all different models using multi-regime approach, we can see that the accuracies of these models are similar. The reason may be because the proposed multi-regime approach decomposes the complicated problem into several simple problems, and as for these simple problems, all these models can achieve good results, so the results are similar. Moreover, this may be the reason why the proposed multi-regime approach can improve the accuracy. It simplifies the complicated multi-regime time series prediction problem.

Second, as for the stability, similar to the experiment results in Section 4.2.1, the stability of the proposed model is better than others except SVR related models. The AREs and RMSEs of SVR related models are nearly deterministic. By comparing the stabilities of models that use the multi-regime approach and models that do not use multi-regime approach, e.g., M-BPNN and BPNN, M-ELM-RF and ELM-RF, it can be observed that the proposed multi-regime approach can also improve the stability. As for the running time, SVR is also the most efficient model, and the efficiency of the proposed approach is also acceptable, while BPNN is too time-consuming.

Last, by comparing two solutions for determining regime probabilities, we can see that they are both effective, and the difference between two solutions is not very large.

In summary, the proposed multi-regime approach is effective in handling multi-regime time series, and it can improve both the accuracy and stability. If the proposed RF-based ELM ensemble model is integrated with the multi-regime approach, then the approach can be accurate, stable and efficient in predicting multi-regime time series.

4.2.3. Application to engine EGT time series prediction

Section 4.2.2 uses the simulated turbofan engine time series, and in this subsection, the proposed model is applied to the prediction of EGT time series of CFM engine, which is acquired from real sensors. EGT is one of the most important performance parameters in a turbofan engine (Fu, Ding, & Zhong, 2009), and the airlines and manufacturers need to evaluate these parameters of turbofan engine to ensure the flight safety, reduce maintenance costs, and increase the aircraft lifetime. EGT is a measure of the engine's efficiency in producing its design level of thrust, and higher EGT can cause more wear of the engine and then the performance of the engine deteriorates (Yılmaz, 2009). Because there is no regime-switching information in this time series, the multi-regime approach will not be used, and only the RF-based ELM ensemble

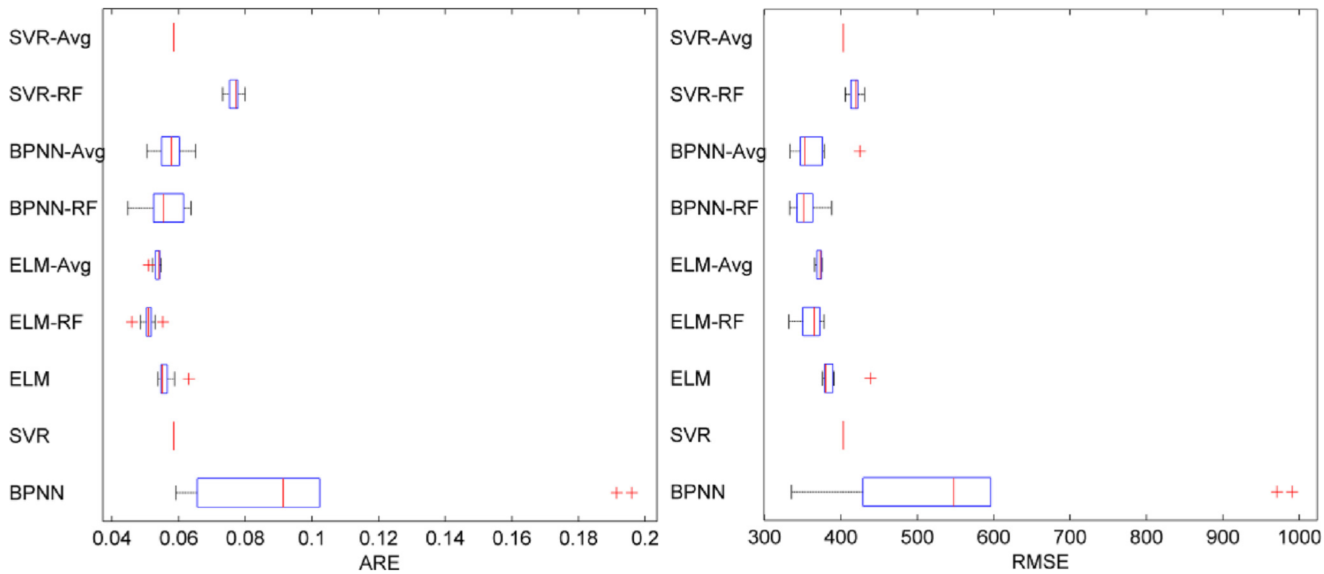


Fig. 7. The boxplots of ARE and RMSE of different models on NN3 time series.

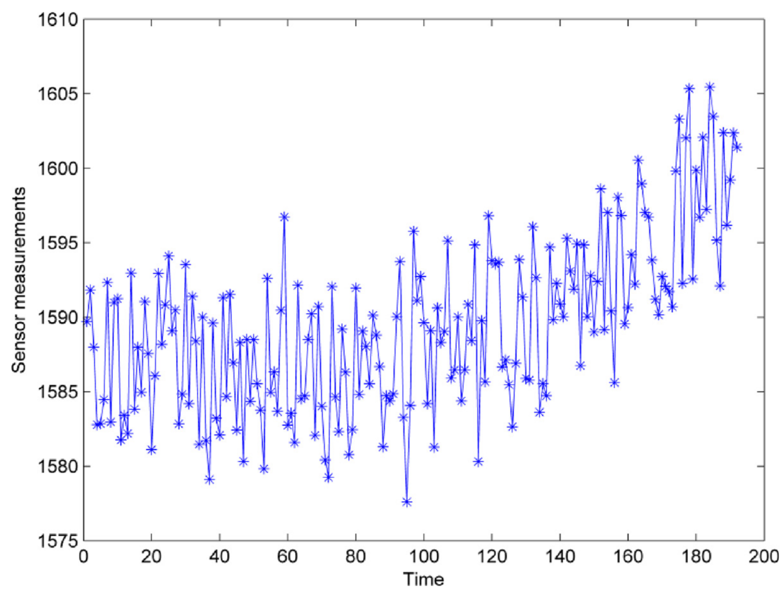


Fig. 8. The FD001 time series.

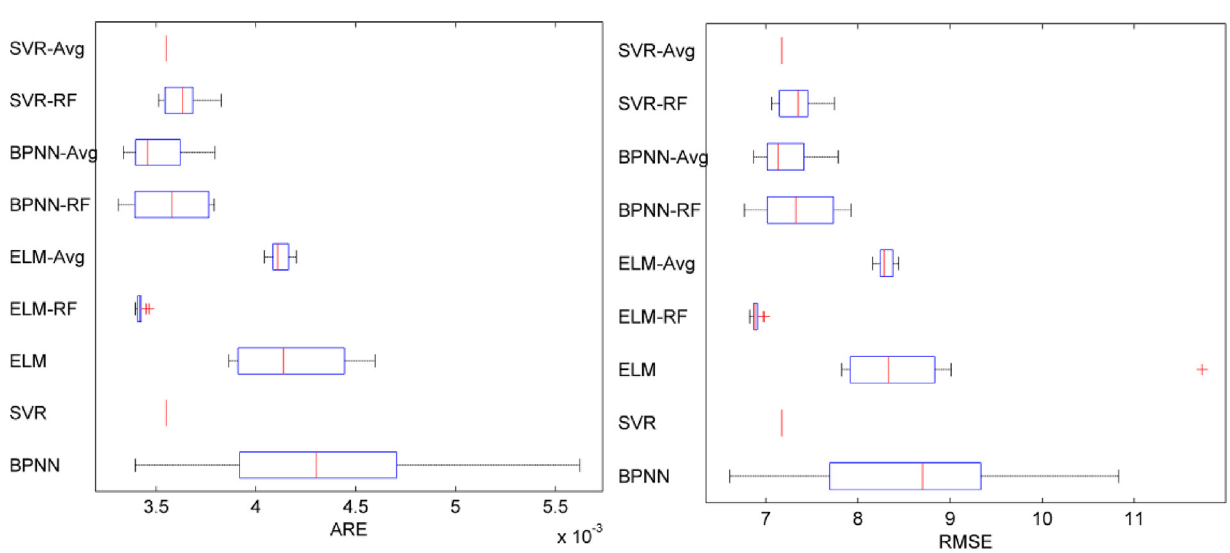


Fig. 9. The boxplots of ARE and RMSE of different models on FD001 time series.

Table 4
The results of different models on FD002 time series when using equiprobability solution.

	ARE	RMSE	Variance (ARE)	Variance (RMSE)	Time (s)
BPNN	0.0843037	145.7774	1.3452e−4	337.2769	5.2619
SVR	0.0709110	118.1823	2.1399e−34	8.9755e−28	***0.0154***
ELM	0.0692168	113.4517	2.0400e−8	0.0942	0.0813
ELM-RF	0.0809038	137.3910	2.7614e−6	7.7917	5.8147
ELM-Avg	0.0691594	113.3563	3.2898e−9	0.0110	2.9714
M-ELM-RF	0.0656726	***104.5766***	6.1912e−10	0.0011	31.6583
M-ELM-Avg	0.0656563	104.6598	5.5704e−10	6.8274e−4	11.3034
M-ELM	***0.0656320***	104.6325	2.5403e−8	0.0604	0.7864
M-BPNN	0.0658146	104.7424	4.2184e−8	0.1366	20.3859
M-SVR	0.0657790	104.6933	***0***	***2.2439e−28***	0.0905
M-BPNN-RF	0.0657380	104.6542	5.3713e−10	0.0015	532.3274
M-BPNN-Avg	0.0657961	104.7138	2.0936e−9	0.0038	510.5044
M-SVR-RF	0.0656924	104.6310	1.3223e−10	7.4796e−05	15.6287
M-SVR-Avg	0.0657790	104.6933	***0***	***2.2439e−28***	1.8314

Table 5
The results of different models on FD002 time series when using frequency-based solution.

	ARE	RMSE	Variance (ARE)	Variance (RMSE)	Time (s)
BPNN	0.0813371	140.4396	1.2436e−4	300.8341	6.8724
SVR	0.0709110	118.1823	***2.1399e−34***	***2.2439e−28***	***0.0195***
ELM	0.0690734	113.2175	4.4182e−8	0.1904	0.0793
ELM-RF	0.0797046	135.9647	4.7736e−6	8.9508	5.5266
ELM-Avg	0.0691220	113.2991	2.7676e−9	0.0093	2.5176
M-ELM-RF	0.0645399	***107.0605***	1.2381e−9	0.0026	22.9991
M-ELM-Avg	***0.0645297***	107.2143	6.2912e−10	0.0019	10.5084
M-ELM	0.0645549	107.2683	1.1502e−8	0.0523	0.4895
M-BPNN	0.0645763	107.1515	3.6288e−8	0.0680	18.3589
M-SVR	0.0646630	107.1499	***2.1399e−34***	***2.2439e−28***	0.0680
M-BPNN-RF	0.0646575	107.2086	4.8887e−9	0.0059	535.9200
M-BPNN-Avg	0.0646577	107.1551	1.4182e−9	0.0055	512.9440
M-SVR-RF	0.0646274	107.1552	6.7822e−11	1.3839e−4	12.8953
M-SVR-Avg	0.0646630	107.1499	***2.1399e−34***	***2.2439e−28***	2.3476

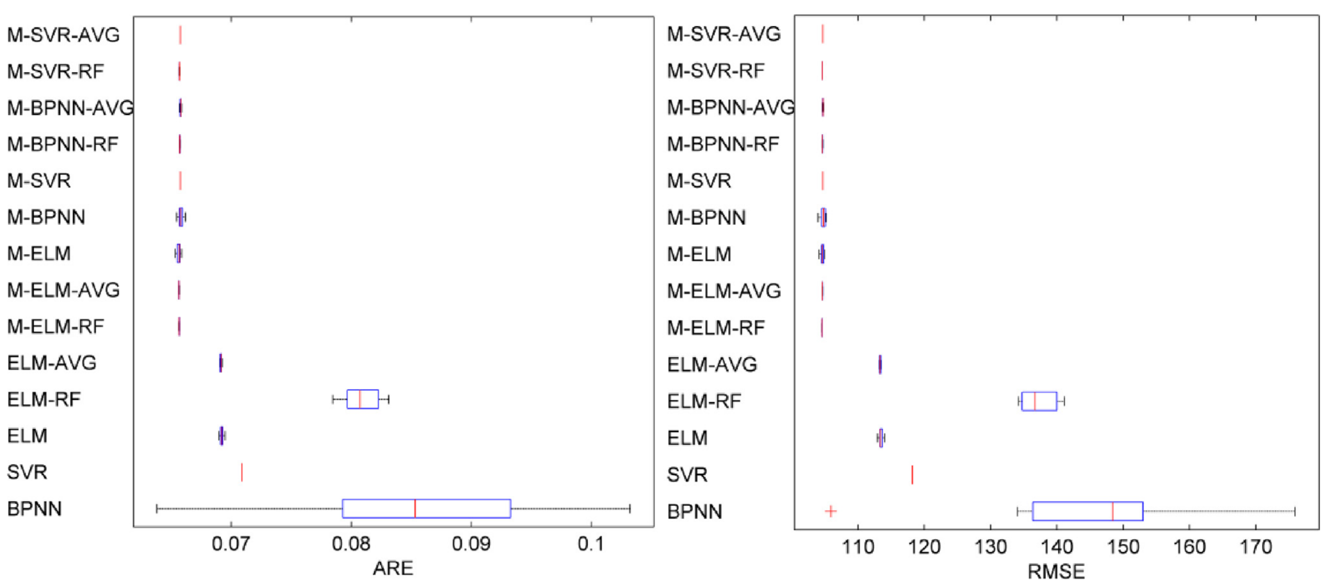


Fig. 10. The boxplots of different models on FD002 time series using equiprobability solution.

model is applied. Similar to NASA turbofan engine time series, noise exists in EGT time series.

There are 1000 points in this time series, the first 600 timestamps are used as training points, and the left 400 points are used to evaluate the models. Similarly, the previous 4 timestamps are used to predict the fifth timestamp; thus, 596 training samples and 396 test samples are obtained. In the experiments, the hidden neuron numbers of BPNN and ELM are all set 30, and the base learner numbers of all ensemble models are set 20. The results are shown in Table 6 and Fig. 11.

From the results, BPNN-RF model is the most accurate one, which demonstrates that the RF-based ensemble strategy is effective. However, BPNN-RF model is very time-consuming. The proposed RF-based ELM ensemble model is better than other models except BPNN-RF model. Moreover, the proposed model is very efficient. As for the stability, the stability of the proposed model is better than others except SVR related models. The AREs and RM-

Table 6
The results of different models on EGT time series.

	ARE	RMSE	Variance (ARE)	Variance (RMSE)	Time (s)
BPNN	0.0915	61.0572	0.0065	2.1941e+3	5.4391
SVR	0.0624	45.6406	***2.1399e−34***	***0***	0.3803
ELM	0.0556	41.0211	4.9901e−8	0.0440	***0.3144***
ELM-RF	0.0529	39.3279	2.7037e−7	0.1549	9.0978
ELM-Avg	0.0555	40.9085	8.7111e−9	0.0064	5.3478
BPNN-RF	***0.0436***	***32.9260***	2.4227e−6	1.5559	1.5418e+3
BPNN-Avg	0.0548	40.4280	6.0309e−5	24.3589	1.0693e+3
SVR-RF	0.0625	45.6776	2.1290e−10	8.5843e−5	30.3161
SVR-Avg	0.0624	45.6406	***2.1399e−34***	***0***	26.3014

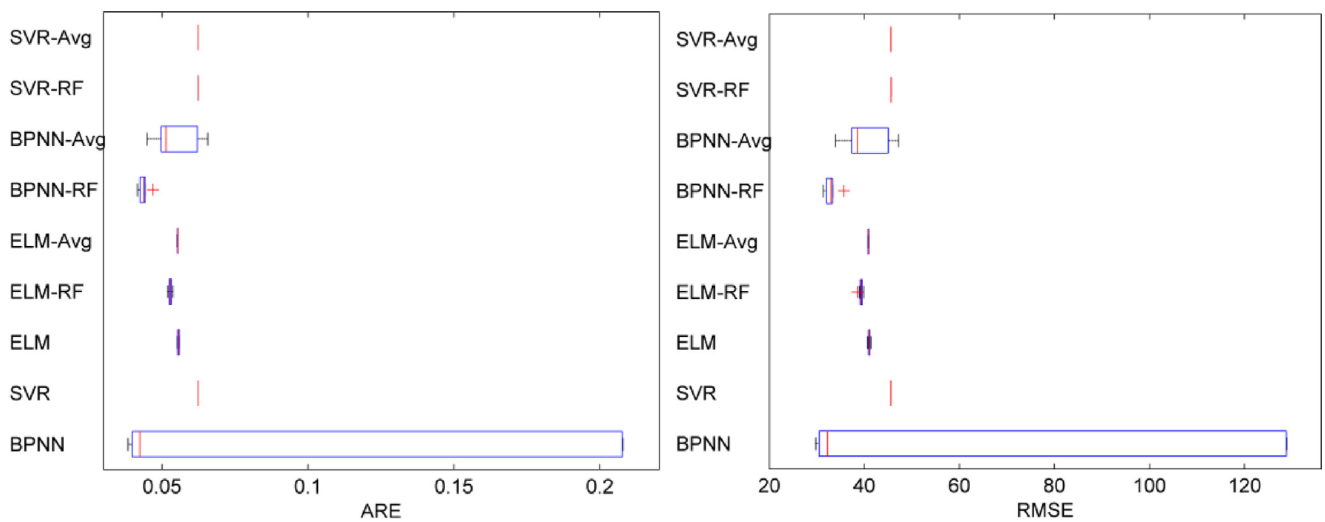


Fig. 11. The boxplots of different models on EGT time series.

SEs of SVR related models are nearly deterministic. Besides, because noise exists in this time series, the variance of BPNN model is very large, i.e., the width of BPNN box is very large in Fig. 11. By using the RF-based ensemble strategy, the stability is improved greatly, which demonstrates that the proposed model can be robust against noise.

In general, taking into account the results of all data, the proposed RF-based ELM ensemble model performs well in accuracy, stability and efficiency. After careful thought, the superiority of the proposed model is due to the combination of RF and ELM. Specifically, bootstrap sampling in RF and both the input weights and all biases which are randomly initialized in ELM make base learners sufficient diversity. And, as for the input weights and numbers of hidden neurons, corresponding output weights are calculated with Eq. (2) uniquely. So, sufficient diversity of the results of ELMs is guaranteed. In addition, bagging ensemble strategy and regularization in ELM make the proposed model better generalization ability.

5. Conclusion

In order to achieve the accuracy, stability and efficiency together in time series prediction, the RF-based ELM ensemble model is proposed. In order to handle multi-regime time series, a multi-regime approach is proposed based on the specific properties of turbofan engine time series. The RF-based ELM ensemble model and the multi-regime approach can be integrated to achieve better performance.

First, the recently proposed ELM is very efficient; thus, it is adopted in the proposed model. In order to overcome overfitting and improve the generalization performance, the regularized ELM and ensemble learning strategy are used. The bootstrap sampling technique is adopted to achieve the diversity of base-level ELM models, and then the RF model is used as the combiner to achieve more accurate and stable performance of the ensemble model. Next, based on the specific properties of turbofan engine time series, a multi-regime approach is proposed to handle it. The regimes are first separated because they are known, and the proposed RF-based ELM ensemble model is used to learn the models of all regimes, individually. All the regime models are then aggregated to predict the future flight cycle, whose operational regime is unknown.

The proposed RF-based ELM ensemble model is evaluated by using NN3 time series, which demonstrates that the proposed model is accurate, stable and efficient. Then the proposed model and multi-regime approach are used in NASA turbofan engine time series prediction, and the results show that the proposed model is effective in predicting time series with one regime, and the approach integrating the RF-based ELM ensemble model and multi-regime approach can be accurate, stable and efficient in predicting multi-regime time series. Last, the proposed RF-based ELM ensemble model is applied to the real EGT time series prediction. The advantages of the proposed approach are threefold: (i) the proposed RF-based ELM ensemble model can achieve accuracy, stability and efficiency together; (ii) the multi-regime approach can improve the accuracy and stability when dealing with multi-regime time series; (iii) the proposed approach can be robust against noise. Noise exists in NASA time series and EGT time series, and the proposed approach can remain good performance even no denoise approach is used to preprocess the data.

Further studies of this paper are listed as follows:

1 This paper mainly focuses on one-step-ahead time series prediction, so multi-step-ahead time series prediction should be studied for achieving good performance.

2 Early warnings technology is the challenge for prediction domain. So, it is urgent to increase the accuracy of early warnings.
3 This paper uses a number of data from sensor to predict the performance parameters of mechanical devices. But, physical structure of equipment is not used, which reflects mechanical correlation. So, data-driven associated with physical model should be studied in the future.
4 In term of the application, the remaining life of the significant equipment obtained from the prediction results deserves to be studied.

Acknowledgment

The authors are thankful for the supports from the National Natural Science Foundation of China, China (No. 51075083), Specialized Research Fund for the Doctoral Program of Higher Education, China (Priority development area) (20112302130003), and the major project in a certain domain, China.

References

Ailliot, P., & Monbet, V. (2012). Markov-switching autoregressive models for wind time series. *Environmental Modelling & Software, 30*, 92–101.
Bartlett, P. L. (1998). The sample complexity of pattern classification with neural networks: The size of the weights is more important than the size of the network. *IEEE Transactions on Information Theory, 44*, 525–536.
Breiman, L. (1996). Bagging predictors. *Machine Learning, 24*, 123–140.
Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5–32.
Cao, J., Lin, Z., Huang, G.-B., & Liu, N. (2012). Voting based extreme learning machine. *Information Sciences, 185*, 66–77.
Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST), 2* 27.
Chen, C.-F., Lai, M.-C., & Yeh, C.-C. (2012). Forecasting tourism demand based on empirical mode decomposition and neural network. *Knowledge-Based Systems, 26*, 281–287.
Chong, A. Y.-L. (2013). Predicting m-commerce adoption determinants: A neural network approach. *Expert Systems with Applications, 40*, 523–530.
das Chagas Moura, M., Zio, E., Lins, I. D., & Droguett, E. (2011). Failure and reliability prediction by support vector machines regression of time series data. *Reliability Engineering & System Safety, 96*, 1527–1534.
Deng, W., Zheng, Q., & Chen, L. (2009). Regularized extreme learning machine. In *Proceedings of the 2009 IEEE Symposium on Computational Intelligence and Data Mining (CIDM'09)* (pp. 389–395).
Frederick, D. K., DeCastro, J. A., & Litt, J. S. (2007). User's guide for the commercial modular aero-propulsion system simulation (C-MAPSS).Nasa/arl.
Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences, 55*, 119–139.
Fu, X., Ding, G., & Zhong, S. (2009). Aeroengine turbine exhaust gas temperature prediction using support vector machines. In *Proceedings of the 2009 International Symposium on Advances in Neural Networks (ISNN)* (pp. 235–241).
Ghanbari, A., Kazemi, S., Mehmanpazir, F., & Nakhostin, M. M. (2013). A cooperative ant colony optimization-genetic algorithm approach for construction of energy demand forecasting knowledge-based expert systems. *Knowledge-Based Systems, 39*, 194–206.
Gheyas, I. A., & Smith, L. S. (2011). A novel neural network ensemble architecture for time series forecasting. *Neurocomputing, 74*, 3855–3864.
Hadavandi, E., Shavandi, H., & Ghanbari, A. (2010). Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting. *Knowledge-Based Systems, 23*, 800–808.
Hamilton, J. D. (2010). Regime switching models, *Macroeconometrics and Time Series Analysis* (pp. 202–209). Palgrave Macmillan UK.
Ho, T. K. (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*, 832–844.
Huang, G.-B., Zhou, H., Ding, X., & Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 42*, 513–529.
Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2004). Extreme learning machine: A new learning scheme of feedforward neural networks. In *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks* (pp. 985–990).
Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Extreme learning machine: Theory and applications. *Neurocomputing, 70*, 489–501.
Huang, G., Huang, G.-B., Song, S., & You, K. (2015). Trends in extreme learning machines: A review. *Neural Networks, 61*, 32–48.
Kim, C.-J., & Nelson, C. R. (1999). *State-space models with regime switching: Classical and Gibbs-sampling approaches with applications*: 1. MIT Press Books.
Lan, Y., Soh, Y. C., & Huang, G.-B. (2009). Ensemble of online sequential extreme learning machine. *Neurocomputing, 72*, 3391–3395.
Lee, Y.-S., & Tong, L.-I. (2011). Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming. *Knowledge-Based Systems, 24*, 66–72.
Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R News, 2*, 18–22.
Liu, C.-H., & Chen, H.-C. (2012). A novel CBR system for numeric prediction. *Information Sciences, 185*, 178–190.
Liu, N., & Wang, H. (2010). Ensemble based extreme learning machine. *IEEE Signal Processing Letters, 17*, 754–757.
Melin, P., Soto, J., Castillo, O., & Soria, J. (2012). A new approach for time series prediction using ensembles of ANFIS models. *Expert Systems with Applications, 39*, 3494–3506.
Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., & Lendasse, A. (2010). OP-ELM: Optimally pruned extreme learning machine. *IEEE Transactions on Neural Networks, 21*, 158–162.
Miche, Y., van Heeswijk, M., Bas, P., Simula, O., & Lendasse, A. (2011). TROP-ELM: A double-regularized ELM using LARS and Tikhonov regularization. *Neurocomputing, 74*, 2413–2421.
Patton, A. J. (2012). A review of copula models for economic time series. *Journal of Multivariate Analysis, 110*, 4–18.
Rémillard, B., Papageorgiou, N., & Soustra, F. (2012). Copula-based semiparametric models for multivariate time series. *Journal of Multivariate Analysis, 110*, 30–42.
Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review, 33*, 1–39.
Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In *Proceedings of the 2008 International Conference on Prognostics and Health Management (PHM)* (pp. 1–9).
Singh, R., & Balasundaram, S. (2007). Application of extreme learning machine method for time series analysis. *International Journal of Intelligent Technology, 2*, 256–262.
Wang, T. (2010). *Trajectory similarity based prediction for remaining useful life estimation*. University of Cincinnati.
Wang, Y., Li, D., Du, Y., & Pan, Z. (2015). Anomaly detection in traffic using $L1$-norm minimization extreme learning machine. *Neurocomputing, 149*, 415–425.
Xia, M., & Wong, W. (2014). A seasonal discrete grey forecasting model for fashion retailing. *Knowledge-Based Systems, 57*, 119–126.
Xie, X., Lin, L., & Zhong, S. (2014). Process Takagi–Sugeno model: A novel approach for handling continuous input and output functions and its application to time series prediction. *Knowledge-Based Systems, 63*, 46–58.
Yılmaz, İ. (2009). Evaluation of the relationship between exhaust gas temperature and operational parameters in CFM56-7B engines. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 223*, 433–440.