Neurocomputing 205 (2016) 53–63

Contents lists available at ScienceDirect

Neurocomputing

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

Direct interval forecasting of wind speed using radial basis function neural networks in a multi-objective optimization framework

Chi Zhang, Haikun Wei *, Liping Xie, Yu Shen, Kanjian Zhang

Key Laboratory of Measurement and Control of CSE, Ministry of Education, School of Automation, Southeast University, Nanjing 210096, PR China

ARTICLE INFO

Article history:
Received 5 October 2015
Received in revised form
10 March 2016
Accepted 28 March 2016
Communicated by Jiahu Qin
Available online 13 May 2016

Keywords:
Prediction interval
Radial basis function neural network
Multi-layer perceptron
Multi-objective genetic algorithm
Wind speed

ABSTRACT

Point predictions of wind speed can hardly be reliable and accurate when the uncertainty level increases in data. Prediction intervals (PIs) provide a solution to quantify the uncertainty associated with point predictions. In this paper, we adopt radial basis function (RBF) neural networks to perform interval forecasting of the future wind speed. A two-step method is proposed to determine the RBF connection weights in a multi-objective optimization framework. In the first step, the centers of the RBF are determined using the *K*-means clustering algorithm and the hidden-output weights of the RBF are pre-trained using the least squares algorithm. In the second step, the hidden-output weights are further adjusted by the non-dominated sorting genetic algorithm-II (NSGA-II), which aims at concurrently minimizing the width and maximizing the coverage probability of the constructed intervals. We test the performance of the proposed method on three real data sets, which are collected from different wind farms in China. The experimental results indicate that the proposed method can provide higher quality PIs than the conventional multi-layer perceptron (MLP) based methods.

1. Introduction

Wind energy, which is a green and renewable energy, has been under large scale development throughout the world over the past decades [1]. It has shown the most rapid and consistent deployment of power generating capacities among various renewable energy sources [2]. However, since wind power generation mainly depends on wind speed, which has the characteristics of intermittency and stochastic fluctuation, the integration of wind power into power systems poses a number of challenges [3]. To ensure safe and reliable operation of the power system, accurate wind speed forecasting is needed [4].

In the literature, many methods have been proposed for the forecasting of wind speed and most of them focus on point predictions. Depending on the mechanism utilized, there are two mainstream methods for the point forecasting of wind speed: physical and statistical methods [4]. The physical methods generate wind speed predictions by solving fluid dynamics and thermodynamics equations numerically [5]. Unlike physical methods, statistical methods usually construct statistical models to predict wind speed based on a number of historical data. Conventional statistical models, especially the autoregressive moving average (ARMA) models and their extended versions, have been widely applied to the short-term wind speed point forecasting [6,7]. Recently, machine learning models, such as artificial neural networks (ANNs) [8,9] and support vector machines (SVMs) [10,11], have been extensively adopted for wind speed point forecasting. Furthermore, more recently, hybrid or combined models, which combine different models together to take advantage of the strengths of each component model, have also been proposed [12,13].

From a practical point of view, it is risky for the decision-makers to develop operational strategies in the power system management purely according to the point forecasts. The accuracy of point forecasts is often unsatisfactorily low due to the uncertainty in real data sets. Thus, it is necessary to conduct interval forecasting, which can provide information about associated uncertainty in the point predictions. Some methods for wind speed interval forecasting have been proposed recently. Jiang et al. [14] proposed a Bayesian structural break model to predict the future wind speed and its intervals. Song et al. [15] employed a Markov-switching model to perform both point and interval forecasting of the future wind speed. Qin et al. [16] proposed a hybrid model based on the cuckoo search optimization and back-propagation neural network to establish wind speed interval forecasts.

Although there are only a few studies on wind speed interval forecasting, the construction of PIs in other application areas has been studied for many years. Traditional approaches for the construction of PIs first generate the point predictions, and then

* Corresponding author.
E-mail address: hkwei@seu.edu.cn (H. Wei).

http://dx.doi.org/10.1016/j.neucom.2016.03.061
0925-2312/

calculate the corresponding PIs according to the quantile analysis of errors with certain prior assumptions [17]. But since these assumptions about the data may not hold in real-world data sets, the constructed PIs will be unreliable and invalid. In recent years, there has been a growing trend of directly generating interval forecasts, without the need of prior assumptions [18,19]. The main idea of these methods is first to represent the lower and upper bounds of the intervals by some predictive model, and then optimize the model coefficients according to the interval quality assessment indices. In the literature, the MLP neural networks are usually employed to generate the lower and upper bounds of PIs [20]. Since the PI-based objective functions are complex, nonlinear and non-differentiable, traditional derivative-based algorithms cannot be used for their minimization (or maximization). The most commonly used methods for adjusting the MLP connection weights are simulated annealing [18], particle swarm optimization [21] and genetic algorithm [20,22].

However, when using MLP neural networks to perform interval forecasting, the initial values of the connection weights are usually generated randomly [20,23]. The effects of connection weight initialization on the final constructed PIs are usually ignored. Moreover, all the connection weights (i.e., both input-hidden and hidden-output weights) are adjusted when training a MLP according to the PI-based objective functions. These connection weights constitute a very large search space, which may complicate the optimization process. To simplify the optimization process, it is interesting to investigate that whether it is possible to fix part of the connection weights, and then just update the rest part of the weights. Furthermore, another major class of feedforward neural network models, RBF networks, which have been widely and successfully used in many other areas, have not been applied to the field of direct interval forecasting.

In view of the above-mentioned issues, RBF networks are proposed in this paper to perform direct interval forecasting of the future wind speed. Note that the assessment indices of PIs, which are used to train the neural networks, are two competing indices (coverage probability and the normalized average width). Since a narrow interval will induce a low coverage probability, whereas wide intervals may be required to obtain high coverage probability. A high quality interval is the one with high coverage probability but small width. Although a combined measure index, coverage width-based criteria [18], is proposed to simplify the optimization problem, the PI construction in essence is a multi-objective optimization problem [20,22]. Thus the interval forecasting of wind speed in this study is performed in a multi-objective optimization framework [22,23]. That is, the RBF is trained to concurrently minimize the width and maximize the coverage probability of the constructed PIs. The multi-objective genetic algorithm, NSGA-II, is used to adjust the RBF weights. Moreover, in order to effectively initialize the connection weights and reduce the dimensionality and complexity of search space, a two-step method is proposed to determine the RBF weights. Specifically, first, the centers of the RBF are determined in an unsupervised manner and the hidden-output weights are pre-trained using the squared-error objective function. Then the values of the centers are fixed, and only the hidden-output weights are further adjusted by NSGA-II using the PI-based objective functions. In order to evaluate the proposed method, three real data sets of hourly mean wind speed measurements are utilized in this paper.

The main contributions of the paper are listed below. (1) RBF neural networks are proposed to perform direct interval forecasting in a multi-objective optimization framework, which is, to the best of our knowledge, the first time to adopt RBF in this field. (2) A two-step method is proposed to determine the RBF connection weights. The first step is to pre-train the RBF using the squared-error objective function and the second step is to further tune the hidden-output weights with the PI-based objective functions. (3) Demonstrated results from three case studies indicate that the proposed approach generates higher quality PIs than the conventional MLP-based methods.

The reminder of the paper is organized as follows. Section 2 introduces the problem formulation for direct interval forecasting. Section 3 presents the proposed method for PI construction. The experimental procedure and results are presented and discussed in Section 4. Finally, conclusions are drawn in Section 5.

2. Problem formulation for direct interval forecasting

2.1. Direct interval forecasting

Traditional methods for the construction of PIs first generate the point forecasts, and then calculate the corresponding PIs according to the quantile analysis of errors with certain prior assumptions [17,24]. As these assumptions may not hold in real-world case studies, the constructed PIs will be unreliable and invalid. In contrast, direct interval forecasting methods generate the lower and upper bounds of PIs directly, without any assumptions about the error distributions. Specifically, given a set of input-target pairs $D=\{(\mathbf{x}_i,t_i), i=1,2,\ldots,N\}$, where t_i is the i-th target and $\mathbf{x}_i$ denotes the relevant inputs, the lower and upper bounds of PIs can be represented by some predictive model as

$$y_l(\mathbf{x}_i)=f(\mathbf{x}_i;\mathbf{w}_l) \tag{1}$$

$$y_u(\mathbf{x}_i)=f(\mathbf{x}_i;\mathbf{w}_u) \tag{2}$$

where $y_l(\mathbf{x}_i)$ and $y_u(\mathbf{x}_i)$ are the i-th lower and upper bounds, respectively, and $\mathbf{w}_l$ and $\mathbf{w}_u$ are the corresponding parameter vectors. A valid PI is composed of lower and upper bounds in which a future unknown value of the target y_0 (corresponding to inputs $\mathbf{x}_0$) is expected to lie with a predetermined coverage probability (confidence level) $(1-\alpha)$, i.e.,

$$P(y_0\in[y_l(\mathbf{x}_0),y_u(\mathbf{x}_0)])=1-\alpha \tag{3}$$

In the literature, the most often used predictive models for direct interval forecasting are MLP neural networks [18,20]. The MLP used for direct interval forecasting usually consists of an input layer, a hidden layer, and an output layer. The output layer is composed of two outputs to directly generate the lower and upper bounds of PIs. The symbolic MLP with two outputs for direct interval forecasting is shown in Fig. 1.

For the MLP with p input neurons, the output value of one hidden neuron z_j is calculated as follows:

$$z_j=\text{sigmoid}\left(\mathbf{w}_j^T\mathbf{x}\right)=\frac{1}{1+\exp[-(\sum_{k=1}^{p}w_{jk}x_k+w_{j0})]},\quad j=1,2,\ldots,h \tag{4}$$

where h is the number of hidden neurons, w_{jk} is the weight from the k-th input neuron to the j-th hidden neuron, w_{j0} is the bias, x_k is the k-th component of the input vector. The outputs y_u and y_l of the MLP are calculated taking the hidden neurons as their inputs

$$y_u=\mathbf{v}_u^T\mathbf{z}=\sum_{m=1}^{h}v_{um}z_m+v_{u0} \tag{5}$$

$$y_l=\mathbf{v}_l^T\mathbf{z}=\sum_{m=1}^{h}v_{lm}z_m+v_{l0} \tag{6}$$

where v_{um} and v_{lm} are the weights from the m-th hidden neuron to the upper and lower bounds outputs, respectively, v_{u0} and v_{l0} are the biases.