Energy Conversion and Management 196 (2019) 1395–1409

Contents lists available at ScienceDirect

Energy Conversion and Management

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

Wind speed forecasting based on Quantile Regression Minimal Gated Memory Network and Kernel Density Estimation

Zhendong Zhang[a], Hui Qin[a,*], Yongqi Liu[a], Liqiang Yao[b], Xiang Yu[c], Jiantao Lu[a], Zhiqiang Jiang[a], Zhongkai Feng[a]

[a] School of Hydropower and Information Engineering, Huazhong University of Science and Technology, Wuhan, Hubei, China
[b] Changjiang River Scientific Research Institute of Changjiang Water Resources Commission, Wuhan, Hubei, China
[c] Provincial Key Laboratory for Water Information Cooperative Sensing and Intelligent Processing, Nanchang Institute of Technology, Nanchang, Jiangxi, China

ARTICLE INFO

Keywords:
Wind speed prediction
Minimal Gated Memory Network
Quantile Regression
Forecast uncertainty

ABSTRACT

As a renewable and clean energy, wind energy plays an important role in easing the increasingly serious energy crisis. However, due to the strong volatility and randomness of wind speed, large-scale integration of wind energy is limited. Therefore, obtaining reliable high-quality wind speed prediction is of great importance for the planning and application of wind energy. The purpose of this study is to develop a hybrid model for short-term wind speed forecasting and quantifying its uncertainty. In this study, Minimal Gated Memory Network is proposed to reduce the training time without significantly decreasing the prediction accuracy. Furthermore, a new hybrid method combining Quantile Regression and Minimal Gated Memory Network is proposed to predict conditional quantile of wind speed. Afterwards, Kernel Density Estimation method is used to estimate wind speed probabilistic density function according to these conditional quantiles of wind speed. In order to make the model show better performance, Maximal Information Coefficient is used to select the feature variables while Genetic Algorithm is used to obtain optimal feature combinations. Finally, the performance of the proposed model is verified by seven state-of-the-art models through four cases in Inner Mongolia, China from five aspects: point prediction accuracy, interval prediction suitability, probability prediction comprehensive performance, forecast reliability and training time. The experimental results show that the proposed model is able to obtain point prediction results with high accuracy, suitable prediction interval and probability distribution function with strong reliability in a relatively short time on the prediction problems of wind speed.

1. Introduction

With the reduction of fossil energy and the increase in environmental problems caused by using it, wind energy has received more and more attention from all over the world as a clean renewable energy [1]. However, due to the fluctuation and randomness of wind speed, the power grid integrated into the wind power becomes unreliable [2]. Therefore, it is very important to obtain reliable and accurate wind speed predictions and quantify the uncertainty of predictions for the utilization and planning of wind power.

Wind speed prediction methods mainly include physical methods and statistical methods [3]. Physical methods usually predict wind speed from physical mechanisms through meteorological simulation, such as numeric weather prediction (NWP) [4]. Andrade et al. [4] used a Grid of NWP method to improve renewable energy forecasting, mainly including wind and solar energy. NWP method usually has high prediction accuracy but a large amount of calculation [5]. Statistical methods first construct feature inputs using historical wind speed and then select appropriate prediction method to predict wind speed [6]. Many machine learning methods are used to predict wind speed. Traditional time series methods include Autoregressive (AR), Moving Average (MA) and Auto Regressive Moving Average (ARMA) are usually used for short-term forecasting of wind speed [7]. These methods are linear and their ability for nonlinear or non-stationary time series prediction is limited. Chen and Yu integrated unscented Kalman filter into Support Vector Regression (SVR) based state-space model in order to precisely update the short-term estimation of wind speed sequence [8]. Since SVR solves the support vector through quadratic programming, when the number of samples is large, the storage and calculation of the matrix in the quadratic programming solution process consumes a large amount of machine memory and computation time. Artificial Neural Networks (ANN) is also a commonly used method for

* Corresponding author at: School of Hydropower and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China.
E-mail address: hqin@hust.edu.cn (H. Qin).

https://doi.org/10.1016/j.enconman.2019.06.024
Received 9 March 2019; Received in revised form 24 April 2019; Accepted 27 April 2019

predicting wind speed since it can describe the nonlinearity of wind speed [9]. Some new proposed machine learning methods are also used to predict wind speed, such as Extreme Learning Machine (ELM) [10]. Due to the rapid development of deep learning in recent years, the performance of many traditional machine learning methods is inferior to that of deep learning methods [11]. Among deep learning methods, Recurrent Neural Networks (RNN) is suitable for solving sequence problems such as wind speed time series [12] since its network structure considers timing information. However, RNN may face long-term dependency problems when the sequence length is too long [13]. Long Short-Term Memory Network (LSTM) is proposed to solve this problem [14]. Since wind speed is time series data, LSTM has been used to predict wind speed [15].

There are two most important variants of LSTM, one is adding Peephole Connections to LSTM [16], and the other is Gated Recurrent Unit (GRU) that simplifies the gate structure of LSTM to reduce training time [17]. Greff et al tested the performance of eight variants of LSTM with three classic cases, and obtained some important conclusions [18]: (1) coupling the input and forget gates, or removing peephole connections simplified LSTM without significantly decreasing performance; (2) the forget gate and the output activation function are the most critical components of the LSTM block. The conclusions of Greff are crucial for designing a more efficient gate structure memory network with better performance. Therefore, the first problem need to solve is how to design a simplest gate structure memory network for wind speed prediction without reducing prediction accuracy. The idea of this paper is that based on Greff's two conclusions, Minimal Gated Memory Network (MGM) is proposed, designing it with as simple structure and few weight variables as possible.

However, these wind speed prediction method mentioned above are all point prediction methods, lacking the ability to quantify forecast uncertainty [19]. A hybrid model based on shared weight LSTM and Gaussian Process Regression (GPR) is proposed for probabilistic wind speed forecasting [20], but this method has a premise of Gaussian assumptions. An ensemble of mixture density ANN is used for probabilistic wind speed forecasting, which can evaluate the uncertainties of model misspecification [21]. Quantile Regression (QR) is used to extend an existing wind power forecasting system with probabilistic forecasts since it can estimate the conditional distribution of the dependent variable [22]. In 1978, Koenker proposed the linear QR model [23]. Obviously, linear QR is difficult to solve the complex non-linear problems such as wind speed prediction. Quantile Regression Neural Network (QRNN) is combined with QR and ANN by Taylor in 2000, which not only can handle the non-linear problems but also can quantify the forecast uncertainty [24]. Recently, QRNN began to be used to obtain the conditional quantile of wind speed and estimate the probability density function (PDF) of wind speed [25]. A probabilistic wind speed forecasting approach based on Deep Belief Network (DBN) and QR is proposed to enhance the model's ability to deal with nonlinearity and quantify the uncertainty of prediction [26]. It can be seen that QR is widely used in wind speed probability prediction. Therefore, the second problem need to solve is how to summarize a hybrid framework combined QR and any point prediction method, which can not only predict wind speed but also quantify the uncertainty of forecasting. In order to take timing information into the model and further improve the accuracy of the forecast, semi-new hybrid model combined QR and LSTM is first proposed using this framework, called QRLSTM. Similarly, another semi-new hybrid model QRGRU is also proposed. Furthermore, a brand new approach combined QR and MGM, called QRMGM, is used to perform probabilistic forecasting.

The prediction results obtained by QR, QRNN, QRLSTM, QRGRU and QRMGM are a series of conditional quantiles of wind speed, which cannot directly get the probability density function (PDF). The PDF of wind speed need to be estimated by probability density estimation methods using these conditional quantiles. Probability density estimation methods can be divided into two categories: parameter estimation methods and non-parametric estimation methods [25]. Kernel Density Estimation (KDE), a classic non-parametric estimation method, does not require a priori assumptions when estimating data distribution, which is the essential difference from the parameter estimation method [25]. Among kernel function of KDE methods, the mean square error of Epanechnikov kernel is optimal [27]. Therefore, KDE is used to estimate the PDF and Epanechnikov kernel function is used as kernel of KDE in this study.

In this study, a new method based on QRMGM and KDE is proposed to forecast wind speed probabilistic density. The main contributions are outlined as follows:

(1) The simplest form of the gated structure memory network, called MGM, is proposed to minimize training time without significantly reducing prediction accuracy.
(2) A framework of hybrid method combining QR and any point prediction models has been summarized. Furthermore, a brand new approach combined QR and MGM, called QRMGM, is used to perform probabilistic forecasting.
(3) Maximal Information Coefficient is used to select the feature variables while Genetic Algorithm is used to obtain optimal feature combinations, which can improve the performance of the model.
(4) Four wind speed prediction cases in Inner Mongolia, China are used to test four methods from five aspects: point prediction accuracy, interval prediction suitability, probability prediction comprehensive performance, forecast reliability and training time. The experimental results show that QRMGM-KDE has the ability to obtain wind speed prediction results with excellent performance on accuracy, uncertainty and reliability.

The remainder of this paper is organized as follows. In Section 2, the related methods are introduced in detail. In Section 3, the evaluation metrics are explained. In Section 4, an application of the proposed methods for wind speed probability density forecasting is presented. In Section 5, the work of this paper is summarized and the conclusions are given. The nomenclatures of this study can be seen in the Table 1.

2. Methods

In this section, Quantile Regression is first reviewed. Then the framework of a hybrid model combining QR and other point prediction models is summarized. After that, a new probabilistic forecasting model is proposed based on this framework. Finally, feature selection and combination methods are introduced to improve the performance of the model.

2.1. Quantile Regression

Regression analysis studies the relationship between the independent variable $X = [x_1, x_2, \cdots, x_n]$, $x_t = [1, x_{t1}, x_{t2}, \cdots, x_{tm}]$ and the conditional expectation of the dependent variable $Y = [y_1, y_2, \cdots, y_n]$ [28]. Quantile Regression (QR) studies the relationship between the independent variable and the conditional quantile of the dependent variable [23]. Traditional regression analysis can only get the central trend of the dependent variable while QR can further infer the conditional probability distribution of the dependent variable. The linear QR model is as follows:

$$Q_{y_t}(\tau|x_t) = f(x_t, \beta(\tau)) = x_t\beta(\tau) \quad t = 1, 2, \cdots, n \tag{1}$$

where $Q_{y_t}(\tau|x_t)$ is the τ-th condition quantile of the dependent variable y_t and $\tau \in (0, 1)$. Regression coefficients $\beta(\tau) = [\beta_0(\tau), \beta_1(\tau), \cdots, \beta_m(\tau)]$. The estimated value $\widehat{\beta}(\tau)$ of $\beta(\tau)$ can be obtained by minimizing the loss function L:

Table 1
Nomenclatures.

X	a set of independent variables	*NWP*	numeric weather prediction
x_t	t-th independent variable	*ARMA*	Auto Regressive Moving Average
Y	a set of dependent variables	*SVR*	Support Vector Regression
y_t	t-th dependent variable	*ANN*	Artificial Neural Networks
τ	quantile	*ELM*	extreme learning machine
$Q_{yt}(\tau\|x_t)$	τ-th condition quantile prediction of the dependent variable y_t	*RNN*	Recurrent Neural Networks
$\beta(\tau)$	regression coefficients of QR under quantile τ	*LSTM*	Long Short-Term Memory Network
$\hat{\beta}(\tau)$	estimated value of $\beta(\tau)$	*GRU*	Gated Recurrent Unit Network
L	loss function	*MGM*	Minimal Gated Memory Network
argmin(L)	function for getting the parameter that minimizes L	*GPR*	Gaussian Process Regression
$\varphi_\tau(u)$	asymmetric function	*AR*	Autoregressive
$\hat{Q}_{y_t}(\tau\|x_t)$	estimated value of $Q_{yt}(\tau\|x_t)$	*DBN*	Deep Belief Network
$Q_{yt}(x_t)$	prediction of the dependent variable y_t	*QR*	Quantile Regression
$f(x_t, \Omega)$	any point prediction model	*QRNN*	Quantile Regression Neural Network
Ω	parameter of any point prediction model	*QRLSTM*	Quantile Regression Long Short-Term Memory Network
$f(x_t, \Omega(\tau))$	hybrid model combining QR and point prediction model $f(x_t, \Omega)$	*QRGRU*	Quantile Regression Gated Recurrent Unit Network
$\Omega(\tau)$	parameter of hybrid model under quantile τ	*QRMGM*	Quantile Regression Minimal Gated Memory Network
$\hat{\Omega}(\tau)$	estimated value of $\Omega(\tau)$	*SVQR*	Support Vector Quantile Regression
$f_t(\tau)$	forget gates in the t-th period under quantile τ	*PDF*	probability density function
$i_t(\tau)$	input gates in the t-th period under quantile τ	*CDF*	cumulative distribution function
$a_t(\tau)$	information state in the t-th period under quantile τ	*KDE*	Kernel Density Estimation
$h_t(\tau)$	output of the hidden layer in the t-th period under quantile τ	*MIC*	Maximal Information Coefficient
m	number of feature input	*GA*	Genetic algorithm
d	number of hidden layer nodes	*RMSE*	root mean square error
$w_h(\tau)$	weight matrix: $d \times d$	*MAPE*	mean absolute percentage error
$w_x(\tau)$	weight matrix: $d \times m$	*CP*	coverage probability
$w_y(\tau)$	weight matrix: $1 \times d$	*MWP*	mean width percentage
$\sigma(x)$	activation function of sigmoid	*MC*	comprehensive metric of interval prediction
$tanh(x)$	activation function of tanh	*CRPS*	continuous ranked probability score
Z_t	a set of samples (condition quantiles) of KDE	*PIT*	probability integral transform
B	bandwidth of KDE	*TT*	training time
$K(x)$	non-negative kernel function		

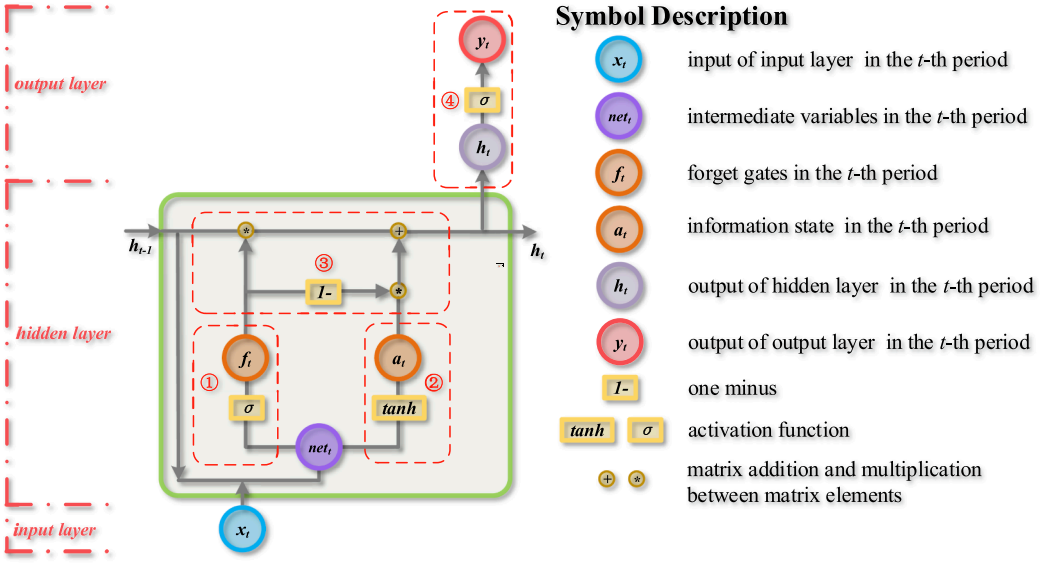


Fig 1. Schematic diagram of QRMGM network structures.

$$\hat{\beta}(\tau) = \arg\min(L); \quad L = \sum_{t=1}^{n} \varphi_\tau(y_t - x_t\beta(\tau)) \tag{2}$$

where $\varphi_\tau(u)$ is an asymmetric function whose formula is as follows:

$$\varphi_\tau(u) = \begin{cases} \tau u & u \geq 0 \\ (\tau - 1)u & u < 0 \end{cases} \tag{3}$$

After that, the τ-th condition quantile of y_t can be estimated by the linear QR model, as follows:

$$\hat{Q}_{y_t}(\tau|x_t) = x_t\hat{\beta}(\tau) \tag{4}$$

2.2. Framework of hybrid model

Through the introduction of line QR, the framework of a hybrid model combining QR and other point prediction models can be summarized as follows:

(1) Suppose $Q_{y_t}(x_t) = f(x_t, \Omega)$ is any point prediction model, where x_t is the model input, Ω is the model parameter and $Q_{y_t}(x_t)$ is the prediction value of y_t.
(2) Then the hybrid model combining QR and this point prediction model is $Q_{y_t}(\tau|x_t) = f(x_t, \Omega(\tau))$. The estimated values $\hat{\Omega}(\tau)$ of $\Omega(\tau)$

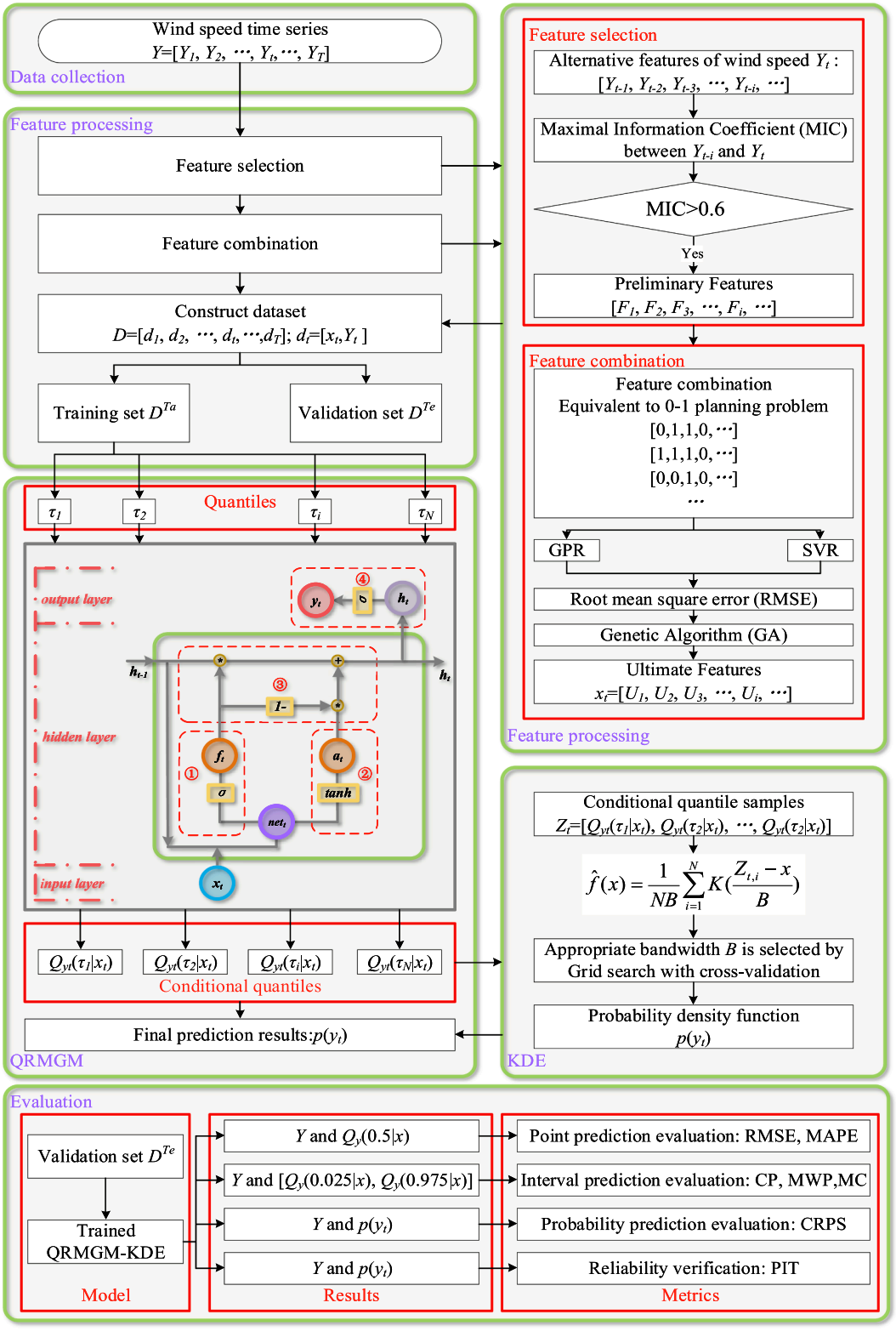


Fig 2. Complete flow chart of QRMGM-KDE.

is obtained by minimizing the loss function $L = \sum_{t=1}^{n} \varphi_\tau(y_t - f(x_t, \Omega(\tau)))$. Sometimes, in order to avoid over-fitting, L1 regularization [29], L2 regularization [29], or joint regularization of L1 and L2 may be added into the loss function.

(3) Finally, the τ-th condition quantile of y_t is estimated by the hybrid model $\hat{Q}_{y_t}(\tau|x_t) = f(x_t, \hat{\Omega}(\tau))$.

The difference between different hybrid models is the calculation of function $f(x_t, \Omega(\tau))$. In order to express unity, the function $f(x_t, \beta(\tau))$ in QR is represented by $f(x_t, \underset{QR}{\Omega}(\tau))$. Using this framework, QR and LSTM, GRU are respectively combined into two semi-new methods QRLSTM, QRGRU. The calculation methods of $f(x_t, \underset{QRLSTM}{\Omega}(\tau))$ and $f(x_t, \underset{QRGRU}{\Omega}(\tau))$ can be found in [14] and [17].

Table 2
Statistical Information of Four Datasets.

Datasets	Time	T	Ta	Tv	min	mean	max	std
unit	1 period = 15 min	period			m/s	m/s	m/s	
Dataset 1	03/25/2016 05:00 – 04/04/2016 14:45	1000	900	100	0.30	6.77	17.88	4.46
Dataset 2	05/08/2016 11:30 – 05/18/2016 21:15	1000	900	100	0.66	9.64	25.39	5.74
Dataset 3	04/20/2016 06:00 – 05/11/2016 01:45	2000	1800	200	0.55	8.45	23.10	5.61
Dataset 4	03/20/2016 00:00 – 04/15/2016 00:45	2500	2250	250	0.30	7.50	21.08	4.74

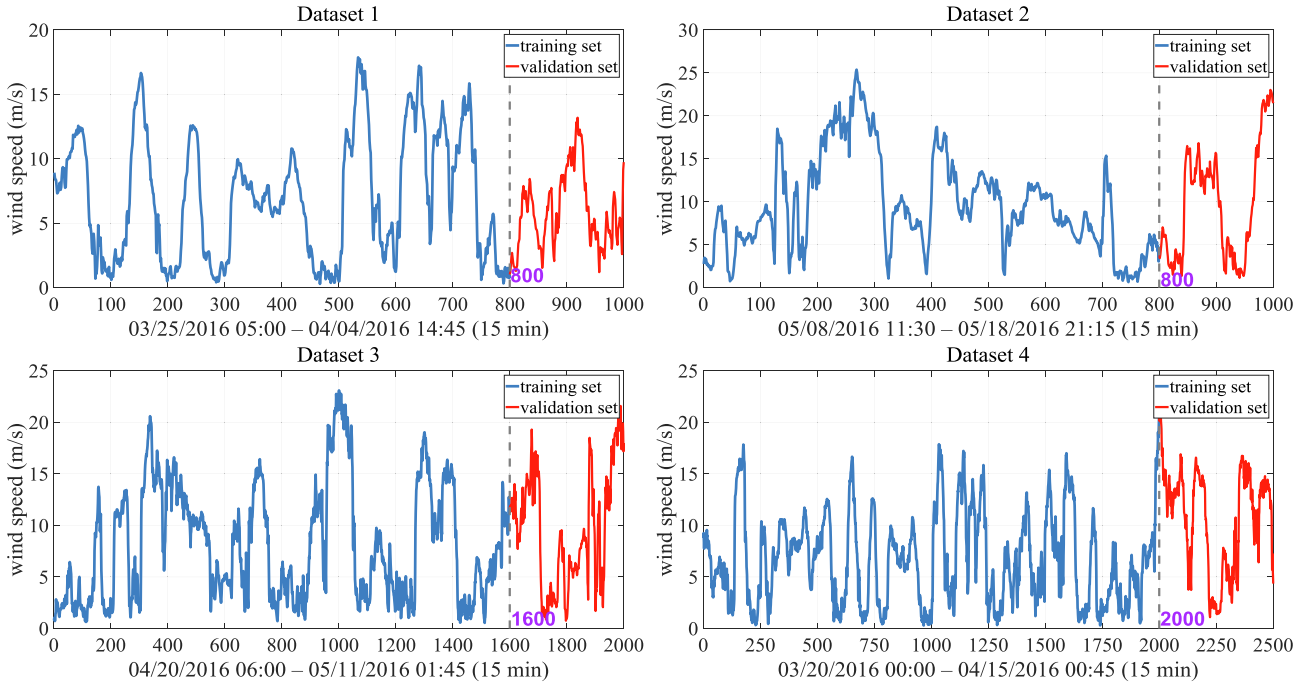


Fig 3. Four datasets of wind speed.

Table 3
The MIC values between the label and alternative feature.

Dataset	y_{t-1}	y_{t-2}	y_{t-3}	y_{t-4}	y_{t-5}	y_{t-6}	y_{t-7}	y_{t-8}	y_{t-9}
1	0.935	0.848	0.760	0.692	0.639	0.592	0.564	0.526	0.495
2	0.935	0.847	0.771	0.708	0.648	0.586	0.550	0.514	0.487
3	0.962	0.885	0.817	0.763	0.718	0.667	0.632	0.597	0.569
4	0.909	0.813	0.748	0.694	0.639	0.613	0.552	0.513	0.471

2.3. Quantile Regression Minimal Gated Memory Network

Minimal Gated Memory Network (MGM) is proposed to simplify the structure of LSTM and reduce the training time without significantly decreasing the prediction accuracy. Further, a hybrid model combined Quantile Regression and Minimal Gated Memory Network is proposed to quantify forecasting uncertainty, which is called QRMGM, as shown in the Fig. 1.

QRMGM still uses the hybrid framework proposed in the Section 2.2, and its core lies in the calculation of $f(x_t, \underset{QRMGM}{\Omega}(\tau))$. The calculation steps of $f(x_t, \underset{QRMGM}{\Omega}(\tau))$ are as follows:

① Calculate forget gates $f_t(\tau)$ and couple input gates $i_t(\tau)$

$$f_t(\tau) = \sigma(net_t(\tau)) = \sigma(w_h(\tau)\cdot h_{t-1}(\tau) + w_x(\tau)\cdot x_t) \tag{5}$$

$$i_t(\tau) = 1 - f_t(\tau) \tag{6}$$

② Calculate current information state $a_t(\tau)$

$$a_t(\tau) = \tanh(net_t(\tau)) = \tanh(w_h(\tau)\cdot h_{t-1}(\tau) + w_x(\tau)\cdot x_t) \tag{7}$$

③ Calculate the output of the hidden layer $h_t(\tau)$

$$h_t(\tau) = f_t(\tau)^* h_{t-1}(\tau) + i_t(\tau)^* a_t(\tau) \tag{8}$$

④ Calculate $f(x_t, \underset{QRMGM}{\Omega}(\tau))$

$$Q_{y_t}(\tau|x_t) = f(x_t, \underset{QRMGM}{\Omega}(\tau)) = \sigma(z_t(\tau)) = \sigma(w_y(\tau)\cdot h_t(\tau)) \tag{9}$$

Suppose that the number of feature input and hidden layer nodes are m and d, then the shape of weight matrix $w_h(\tau)$, $w_x(\tau)$ and $w_y(\tau)$ are $[d \times d]$, $[d \times m]$ and $[1 \times d]$ respectively. $\underset{QRMGM}{\Omega}(\tau)$ represents all weight matrix. $net_t(\tau)$ is the intermediate variable. The symbol $\cdot$ indicates matrix multiplication and the symbol * indicates multiplication between matrix elements. $\sigma(\cdot)$ and $\tanh(\cdot)$ are activation function of *sigmoid* and *tanh* [14].

According to the network structure of LSTM [14] and GRU [17], LSTM has four sets of weight matrix like $[w_h(\tau), w_x(\tau)]$ and GRU has three sets in the hidden layer. MGM has only one set of weight matrix in the hidden layer, which shows that MGM is the simplest form of the gated structure memory network. In MGM, compared with LSTM, the

Table 4
The MIC values between the label and alternative feature.

Dataset	Top 3	Historical wind speed							RMSE(m/s)	
		y_{t-1}	y_{t-2}	y_{t-3}	y_{t-4}	y_{t-5}	y_{t-6}	y_{t-7}	SVR	GPR
Dataset 1	1	√	√	√	×	×	×	×	0.66	0.69
	2	√	√	√	√	×	×	×	0.67	0.69
	3	√	√	√	×	√	×	×	0.73	0.81
Dataset 2	1	√	√	√	√	×	×	×	1.24	1.10
	2	√	√	√	√	√	×	×	1.33	1.13
	3	√	√	√	×	√	×	×	1.48	1.21
Dataset 3	1	√	√	√	√	√	√	×	1.17	1.08
	2	√	√	√	√	×	√	√	1.22	1.12
	3	√	√	√	√	×	×	√	1.37	1.27
Dataset 4	1	√	√	√	×	×	×	×	1.25	1.12
	2	√	√	√	√	×	×	×	1.28	1.17
	3	√	√	√	√	√	×	×	1.43	1.30

input gates and forget gates are coupled, the output gates and bias are removed, two activation functions *sigmoid* and *tanh* are preserved, all of which are consistent with the conclusions of the paper [18], which means that its prediction accuracy will not be significantly reduced. Therefore, in theory, the design of the MGM simplifies the structure of the LSTM and does not significantly reduce the prediction accuracy.

2.4. Kernel density estimation

The hybrid model combined QR can only get the conditional quantile of the prediction, but not the probability density function (PDF) directly. PDF of prediction is obtained by Kernel Density Estimation (KDE) [26] since it is a classic non-parametric estimation method and does not require a priori assumptions. N quantiles are equally spaced from 0 to 1 for τ, which means $\tau = [\tau_1, \tau_2, \cdots, \tau_N]$. For each τ_i, $Q_{y_t}(\tau_i|x_t)$ is obtained by QRMGM. These conditional quantiles constitute a set of samples $Z_t = [Q_{y_t}(\tau_1|x_t), Q_{y_t}(\tau_2|x_t), \cdots, Q_{y_t}(\tau_N|x_t)]$ whose probability density function (PDF) is estimated by KDE. The KDE of the samples Z_t is defined by the equation:

$$\hat{f}(x) = \frac{1}{NB}\sum_{i=1}^{N} K\left(\frac{Z_{t,i} - x}{B}\right) \tag{10}$$

where $B > 0$ is the bandwidth. N is the total number of samples. $K(\cdot)$ is a non-negative kernel function. Epanechnikov kernel is used in this study [27], whose formula is shown as follows:

$$K(\alpha) = \begin{cases} \frac{3}{4}(1-\alpha^2) & \alpha \in [-1, 1] \\ 0 & \alpha \notin [-1, 1] \end{cases} \tag{11}$$

The bandwidth B is one of the most important parameter of KDE. Too wide bandwidth leads to the bias of the estimator while too narrow bandwidth leads to the noise of the estimator [30]. Grid search with cross-validation is used to select an appropriate bandwidth [31].

2.5. Feature selection and combination

(1) Feature selection

Maximal Information Coefficient (MIC) [32], proposed by Reshef in 2011, can be used to measure linear or non-linear correlation between independent variable and dependent variable. Since MIC can capture more extensive relationship instead of specific relationship, such as linear, exponential, periodic relationships and so on [32], MIC is well suited for exploring correlation factors for variables with strong volatility and randomness, such as wind speed. The range of MIC is [0, 1]. The larger the MIC, the greater the correlation. In feature selection, the alternative feature whose MIC value with wind speed of current period (also called label in deep learning) is greater than 0.6 may be left as a feature. Its calculation formula is as follows:

$$MIC(D, X, Y) = \max_{XY<B(|D|)} \frac{I^*(D, X, Y)}{\log_2 \min\{X, Y\}} = \max_{XY<B(|D|)} \frac{\max_G I(D|G)}{\log_2 \min\{X, Y\}} \tag{12}$$

where X and Y are independent variable and dependent variable, respectively. D is a set of ordered pairs. For a grid G, $D|G$ is the probability distribution induced by the data D on the cells of G. $I(D|G)$ denotes mutual information. Function $B(n) = n^{0.6}$ and $|D|$ is the size of data D. Fortunately, MATLAB's extension function *mine(X,Y)* can help solve

Table 5
Comparison model parameter settings.

Model	Symbol	Meaning	Value	Reason
QRMGM	m	number of input layer nodes	–	number of features
QRLSTM	d	number of hidden layer nodes	32	common value [2,4,8,16,32,…]
QRGRU	e	number of hidden layer	3	same
QRNN	η	**decayed learning rate parameter:** initial learning rate	0.01	common value [0.005,0.01,0.05,0.1,…]
	dr	**decayed learning rate parameter:** decay rate	0.9	common value [0.8,0.85,0.9,…]
	ds	**decayed learning rate parameter:** decay steps	15	common value [5,10,15,20,…]
	η_{min}	**decayed learning rate parameter:** minimum of learning rate	1.00E-04	a small value
	bs	**mini-batch parameter:** batch size	32	common value [8,16,32,50,100,…]
	Ep	**mini-batch parameter:** epochs of training	100	converged
	dp	**dropout parameter:** dropout rate	0.2	common value [0.1,0.2,0.3,0.4,…]
	r	**L2 regularization parameter:** penalty parameters	0.01	common value [0.05,0.1,0.15,…]
SVQR	*kernel*	kernel function	Radial Basis Function	a competitive kernel functions
	C	parameter in Radial Basis Function	1	obtained by GA in [-5,5]
GPR	*kernel*	kernel function	Gaussian Function	a competitive kernel functions
	p_1	parameter in Gaussian Function	2	obtained by GA in [-5,5]
	p_2	parameter in Gaussian Function	1	obtained by GA in [-5,5]
QR	*kernel*	kernel function	Gaussian Function	a competitive kernel functions
KDE	K	K-fold cross validation in grid search for KDE's bandwidth	5	same
	B	bandwidth range for KDE in grid search	(0, 10, 0.5)	same
	τ	a set of quantiles	(0, 1, 0.005)	same

Table 6
Point prediction evaluation metrics.

	Datasets	Dataset 1			Dataset 2			Dataset 3			Dataset 4		
	metric	RMSE	MAPE	TT	RMSE	MAPE	TT	RMSE	MAPE	TT	RMSE	MAPE	TT
model	unit	(m/s)	(%)	(s)	(m/s)	(%)	(s)	(m/s)	(%)	(s)	(m/s)	(%)	(s)
QRMGM	min	0.33	4.15	50.1	0.56	4.16	55.1	0.58	4.29	82.4	0.63	4.44	103.3
	mean	0.39	4.69	50.7	0.64	4.45	56.1	0.63	4.60	84.8	0.67	4.65	104.4
	max	0.43	5.06	51.8	0.72	4.77	57.2	0.68	4.80	86.9	0.73	4.97	106.7
QRLSTM	min	0.33	4.13	135.9	0.57	4.26	148.2	0.59	4.36	223.4	0.66	4.53	284.0
	mean	0.39	4.66	139.7	0.67	4.66	150.2	0.64	4.66	226.9	0.68	4.66	289.6
	max	0.42	5.22	143.9	0.77	5.03	154.6	0.70	4.87	231.9	0.69	4.82	296.7
QRGRU	min	0.41	5.00	88.8	0.71	4.69	95.1	0.75	5.31	151.1	0.76	5.21	183.5
	mean	0.45	5.52	90.9	0.79	5.37	97.5	0.79	5.68	154.9	0.82	5.61	187.9
	max	0.49	6.32	93.2	0.91	5.84	99.4	0.82	6.01	159.4	0.86	6.01	192.2
QRNN	min	0.62	7.47	53.7	1.08	7.31	59.6	1.01	7.38	93.5	1.12	7.71	113.2
	mean	0.66	8.06	55.3	1.21	8.08	61.4	1.14	8.07	95.7	1.17	8.13	115.8
	max	0.73	8.64	56.6	1.31	8.69	62.8	1.26	8.64	98.3	1.22	8.46	118.8
SVQR	mean	0.66	8.22	-	1.24	8.34	-	1.17	8.31	-	1.25	8.41	-
GPR	mean	0.69	8.56	-	1.10	7.33	-	1.08	8.21	-	1.12	8.04	-
QR	mean	0.90	11.14	-	1.36	10.98	-	1.62	11.46	-	1.57	10.70	-

MIC and the latest version can be download from the website[1].

(2) Feature combination

Although MIC can evaluate the quality of a single feature, the feature input of the actual wind speed prediction is often a combination of features. In this study, the alternative features with MIC greater than 0.6 are first screened, which are then used for feature combinations. These features have two possibilities of being left or deleted, which corresponds to the 0–1 planning problem. Genetic algorithm (GA) is used to optimize feature combinations [33]. For each feature combination, GPR and SVR are used to predict wind speed and the root mean square error (RMSE) between the predictions and labels is calculated as fitness. The reason for using GPR and SVR to predict wind speed in feature combination is that the calculation of these models is short-lived. It should be noted that feature combination optimization and neural network (such as LSTM, MGM) parameter training are both an iterative optimization problem. The training of neural network is a relatively time consuming process. If the neural network method is used to calculate the RMSE, this would be a two-layer optimization problem and very time-consuming. The complete flow chart of the proposed method QRMGM-KDE is shown in the Fig. 2.

3. Method evaluation metric

In this section, evaluation metrics are explained, including point prediction metrics, interval prediction metrics, probabilistic prediction metrics and reliability metrics.

3.1. Evaluation metric of point prediction

In order to evaluate the accuracy of point prediction, root mean square error (RMSE) [6] and mean absolute percentage error (MAPE) [6] are used. They evaluate the deviation between the predictions and observations. The smaller these metrics, the higher the point prediction accuracy. These formulas are as follows:

$$RMSE = \sqrt{\frac{1}{Tv}\sum_{t=1}^{Tv}(y_t - Y_t)^2} \tag{13}$$

$$MAPE = \frac{1}{Tv}\sum_{t=1}^{Tv}\left|\frac{y_t - Y_t}{Y_t}\right| \times 100\% \tag{14}$$

where y_t and Y_t are prediction and observation, respectively. Tv is the size of validation set sample.

3.2. Evaluation metric of interval prediction

In order to evaluate the suitability of interval prediction, coverage probability (CP) and mean width percentage (MWP) are used in this paper. CP_α [34] is defined as the probability that the observation falls within the prediction interval under confidence level of α. MWP_α [34] is used to measure the prediction interval width. If the interval is wide enough, it is easy to satisfy $CP_\alpha = 100\%$. Such interval is too conservative and does not provide effective information on the uncertainty of the prediction. The ideal prediction interval should have high CP_α and low MWP_α, so the comprehensive metric of interval prediction is defined as MC_α. The smaller the MC_α, the more suitable the prediction interval. These formulas are as follows:

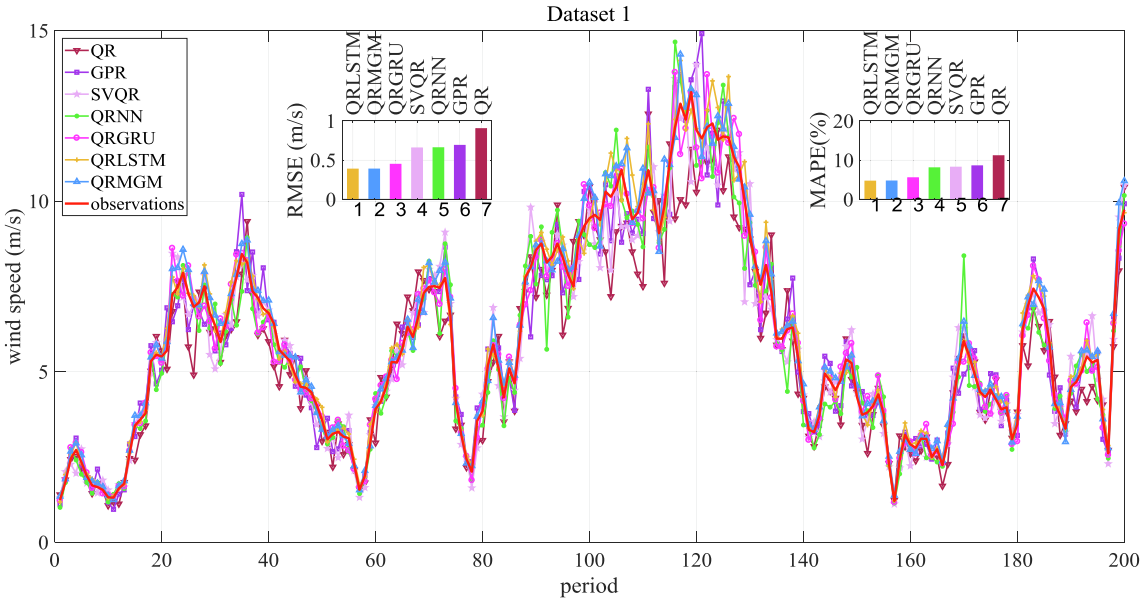
$$CP_\alpha = \frac{c_\alpha}{Tv} \times 100\% \tag{15}$$

$$MWP_\alpha = \frac{1}{Tv}\sum_{t=1}^{Tv}\frac{up_t - down_t}{Y_t} \tag{16}$$

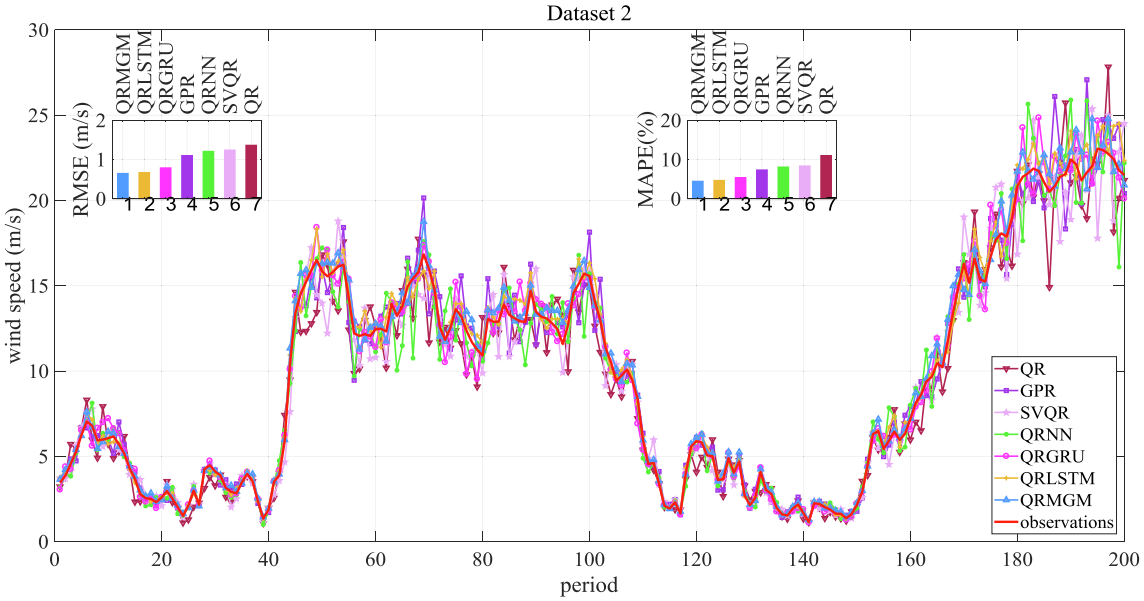
$$MC_\alpha = MWP_\alpha / CP_\alpha \tag{17}$$

where c_α is the number of samples whose observation fall within the prediction interval. up_t and $down_t$ are upper and lower limits of prediction interval.

[1] https://github.com/minepy/minepy/releases.



(a) Point prediction results comparison in Dataset 1



(b) Point prediction results comparison in Dataset 2

Fig 4. Point prediction results comparison.

3.3. Evaluation metric of probability prediction

In order to evaluate the comprehensive performance of probability prediction, continuous ranked probability score (CRPS) is used in this paper [35]. The smaller the CRPS, the better the comprehensive performance of probability prediction. The formula of CRPS is as follows:

$$CRPS = \frac{1}{Tv}\sum_{t=1}^{Tv}\int_{-\infty}^{+\infty}[F(y_t) - H(y_t - Y_t)]^2 dy_t \tag{18}$$

$$F(y_t) = \int_{-\infty}^{y_t} p(x)dx \tag{19}$$

$$H(y_t - Y_t) = \begin{cases} 0 & y_t < Y_t \\ 1 & y_t \geq Y_t \end{cases} \tag{20}$$

where $p(y_t)$ is the PDF of y_t and $F(y_t)$ is its cumulative distribution function (CDF). $H(y_t - Y_t)$ is the Heaviside function.

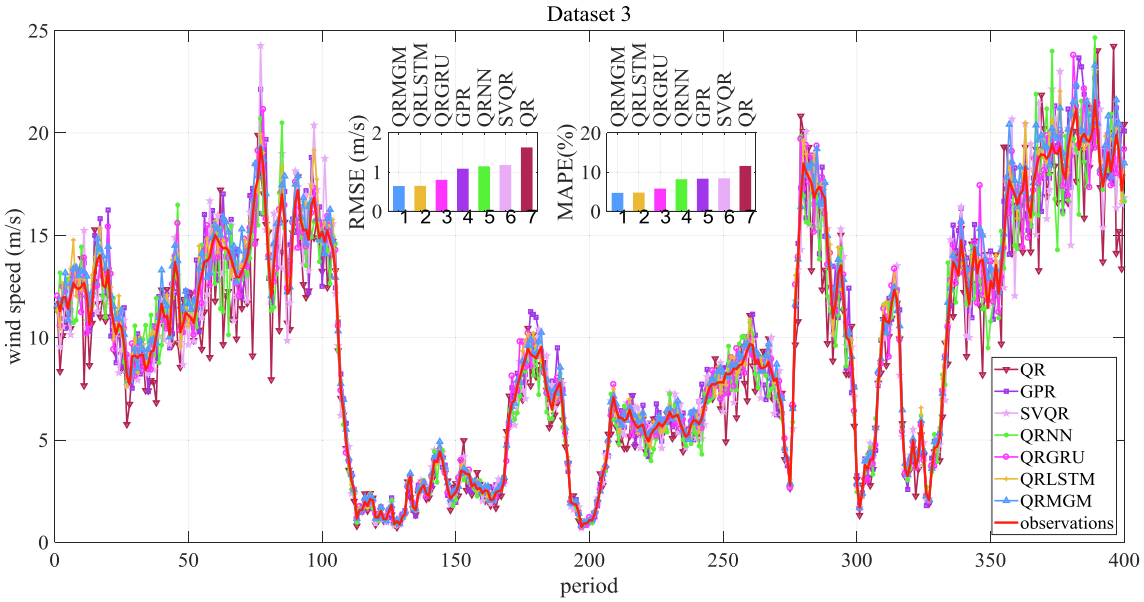
3.4. Evaluation metric of reliability

Reliability refers to the statistical consistency of predictions and observations. Probability integral transform (PIT) values can be used to indicate whether the forecast distribution are excessively high or low, wide or narrow [36]. If the PIT values are subject to a uniform distribution between 0 and 1, the probability prediction is reliable [36]. PIT is calculated from CDF and observation, as follows.

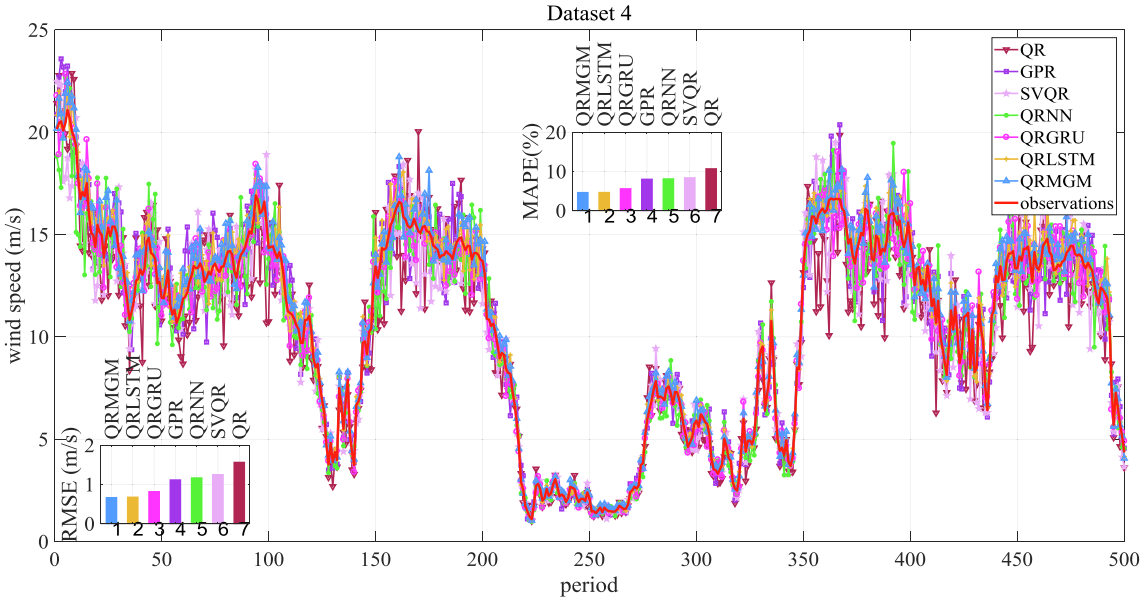
$$PIT = F(Y_t) = \int_{-\infty}^{Y_t} p(x)dx \tag{21}$$

4. Case study

In this section, the dataset introduction is first described. Then the feature inputs are selected and the feature combination is optimized. Next, the experimental design and parameter settings are introduced.



(c) Point prediction results comparison in Dataset 3



(d) Point prediction results comparison in Dataset 4.

Fig 4. (*continued*)

Finally, the performance of the proposed method is verified from five aspects: point prediction accuracy, interval prediction suitability, probability prediction comprehensive performance, forecast reliability and training time.

4.1. Dataset introduction

This study focuses on the wind speed data from the wind farm in Inner Mongolia, China. Four datasets with different time and data lengths are used to test the performance of the model, whose statistical information is shown in the Table 2. *T*, *Ta* and *Tv* represent the size of total sample, training set sample and validation set sample. The minimum, mean, maximum and standard deviation of total sample are abbreviated as min, mean, max and std. Ninety percent of each dataset is used as training set and the rest is used as validation set. The length of a period is 15 min. Four datasets of wind speed are shown in the Fig. 3.

There are four tasks need to be completed in this case study, as follows:

Task I: Select features and optimize feature combinations for wind speed prediction.

Task II: Compare QRMGM with the state-of-the-art wind speed prediction methods from four aspects: point prediction accuracy, interval prediction suitability, probability prediction comprehensive performance and training time.

TaskIII: Verify the reliability of QRMGM.

Task IV: Show probabilistic forecast results of wind speed prediction obtained by QRMGM.

4.2. Task I: Select features and optimize feature combinations

(1) Select feature for wind speed prediction

Table 7
Interval prediction evaluation metrics.

	Datasets	Dataset 1			Dataset 2			Dataset 3			Dataset 4		
model	metric	$CP_{95\%}$	$MWP_{95\%}$	$MC_{95\%}$	$CP_{95\%}$	$MWP_{95\%}$	$MC_{95\%}$	$CP_{95\%}$	$MWP_{95\%}$	$MC_{95\%}$	$CP_{95\%}$	$MWP_{95\%}$	$MC_{95\%}$
QRMGM	min	0.920	0.363	0.375	0.930	0.569	0.585	0.940	0.516	0.529	0.932	0.411	0.427
	mean	0.952	0.365	0.383	0.953	0.570	0.599	0.958	0.518	0.540	0.952	0.412	0.433
	max	0.975	0.366	0.396	0.975	0.572	0.612	0.978	0.519	0.551	0.962	0.413	0.443
QRLSTM	min	0.915	0.364	0.375	0.925	0.567	0.582	0.928	0.516	0.536	0.940	0.411	0.427
	mean	0.948	0.365	0.385	0.957	0.570	0.596	0.953	0.517	0.543	0.949	0.411	0.434
	max	0.975	0.366	0.400	0.980	0.573	0.617	0.963	0.519	0.558	0.964	0.412	0.438
QRGRU	min	0.925	0.401	0.418	0.935	0.696	0.707	0.933	0.609	0.637	0.944	0.509	0.522
	mean	0.943	0.403	0.427	0.961	0.698	0.727	0.946	0.611	0.646	0.957	0.510	0.533
	max	0.960	0.405	0.438	0.990	0.700	0.749	0.958	0.613	0.655	0.978	0.510	0.540
QRNN	min	0.925	0.567	0.600	0.900	0.858	0.908	0.915	0.825	0.860	0.924	0.682	0.717
	mean	0.934	0.569	0.609	0.926	0.861	0.930	0.936	0.828	0.884	0.942	0.683	0.725
	max	0.945	0.571	0.615	0.945	0.865	0.961	0.960	0.830	0.907	0.952	0.685	0.738
SVQR	mean	0.955	0.625	0.654	0.920	0.926	1.007	0.940	0.860	0.915	0.930	0.684	0.735
GPR	mean	0.935	0.569	0.608	0.955	0.886	0.928	0.968	0.891	0.921	0.956	0.682	0.713
QR	mean	0.925	0.739	0.798	0.975	1.368	1.403	0.933	1.174	1.259	0.952	0.931	0.978

In order to further improve the prediction accuracy of the model, MIC is used to select features. In each dataset, the wind speed of the current period is taken as label and the historical wind speed of the previous nine periods is taken as an alternative feature. The MIC values between the label and alternative feature are calculated as shown in the Table 3. The alternative features with MIC greater than 0.6 are highlighted with gray fills, which are left for further feature combinations. In Dataset 1 and 2, the wind speed of previous five periods participates in the feature combination. In Dataset 3 and 4, the wind speed of previous seven and six periods participates in the feature combination, respectively.

(2) Optimize feature combination for wind speed prediction

For each dataset, the method introduced in Section 2.5 is used to optimize feature combination for wind speed prediction. Top 3 feature combinations of the four datasets are listed in the Table 4. The optimal feature combination is highlighted with gray fill. In the comparison experiments, all models use the Top 1 feature combination as input. The symbol √ indicates that the feature is selected while the symbol × indicates that the feature is deleted.

4.3. Comparison design and parameter settings

In order to fully verify the performance of the proposed method, QRLSTM, QRGRU, QRNN [25], SVQR (Support Vector Quantile Regression) [8], GPR [20] and QR [22] are compared with QRMGM from five aspects: point prediction accuracy, interval prediction suitability, probability prediction comprehensive performance, forecast reliability and training time. To make the four neural network methods (QRMGM, QRLSTM, QRGRU, QRNN) play better performance and prevent overfitting, decayed learning rate [37], mini-batch mechanism [38], L2 regularization [29], dropout [39] and multiple hidden layers are added into the method. To make other methods (SVQR, GPR, QR) play better performance, these methods' parameters are optimized by GA. For the fairness of comparison, the same parameter in other method are set to be the same. All model parameters are set to some common values, either optimized by GA or set to the same, as shown in the Table 5. The weight and bias of four neural network methods are all trained by Adam optimization algorithm [40]. Since there are randomness in the training of four neural network methods, these methods run 10 times and the averages are taken as the final results.

4.4. Task II: Compare different methods

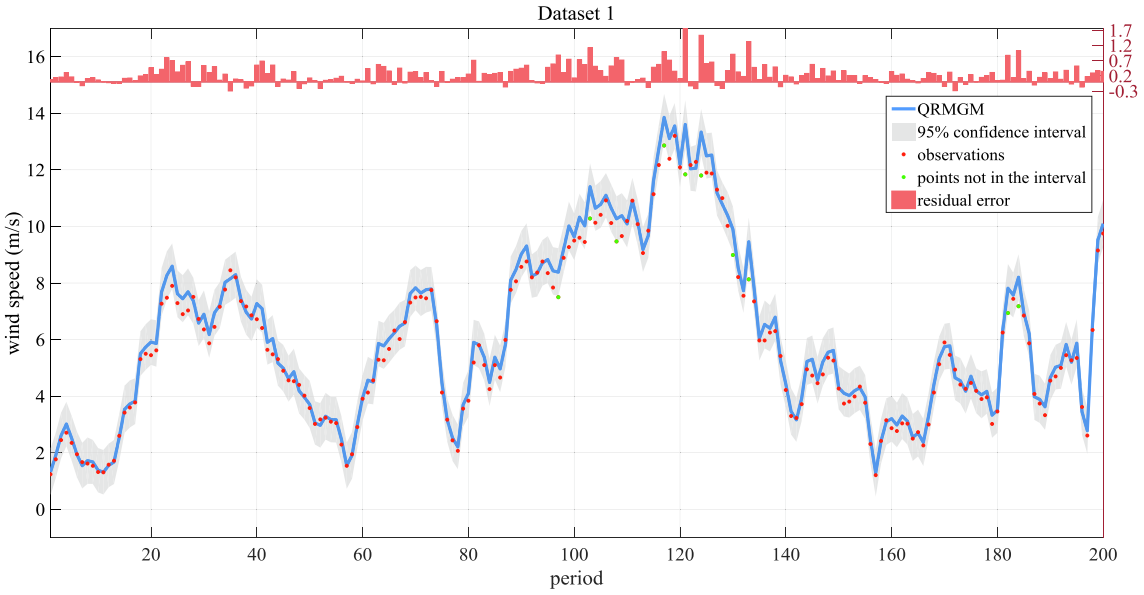
(1) Point prediction results evaluation

The point prediction results evaluation is to verify the prediction accuracy of QRMGM and compare the training time (TT) of the four neural network methods. The point prediction evaluation metrics of seven methods on four datasets are shown in the Table 6. The best metrics in each dataset are highlighted with gray fill. The following three results can be analyzed from the Table 6:

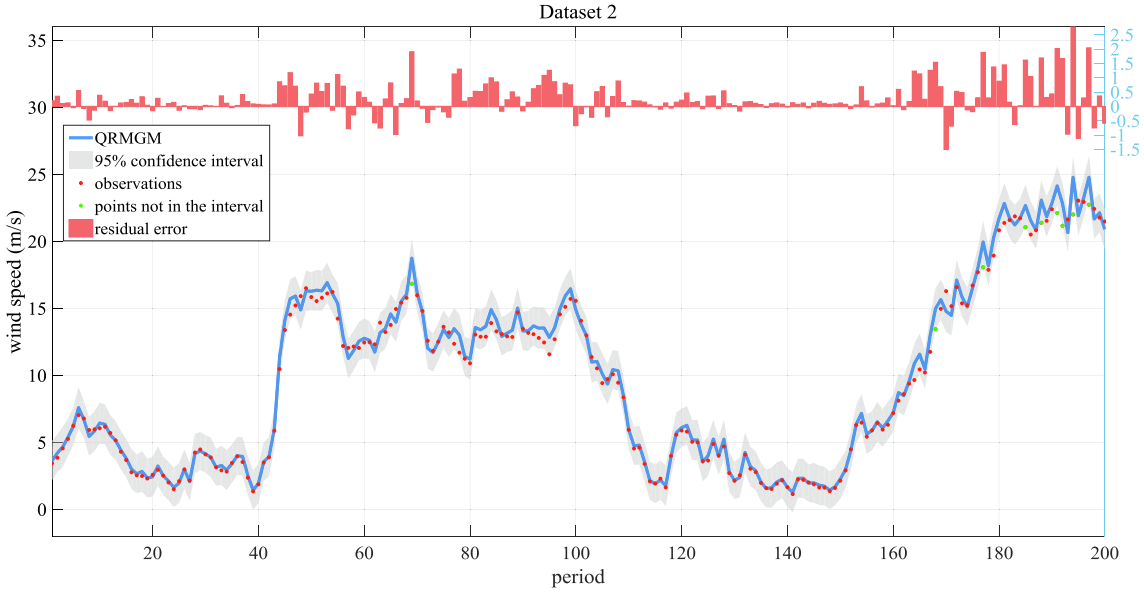
① In the Dataset 1, the mean RMSE values of QRMGM, QRLSTM and QRGRU are 0.39 m/s, 0.39 m/s and 0.45 m/s while the mean RMSE values of QRNN, SVQR, GPR and QR are 0.66 m/s, 0.66 m/s, 0.69 m/s and 0.90 m/s, respectively. It indicates that the prediction accuracy of the three methods (QRMGM, QRLSTM, QRGRU) is significantly higher than the four methods (QRNN, SVQR, GPR, QR) in the Dataset 1. The same conclusion can be obtained by comparing the point prediction evaluation metrics of the remaining three datasets. Since wind speed is time series data, the former three methods can consider the time series information while the latter four methods cannot, which is the reason why the former has higher prediction accuracy than the latter.

② Comparing QRMGM, QRLSTM and QRGRU in more detail, the RMSE and MAPE values of QRLSTM in the Dataset 1 are 0.39 m/s and 4.66%, which are the smallest metrics, indicating that QRLSTM has the highest prediction accuracy in the Dataset 1. In the Dataset 2, the RMSE and MAPE values of QRMGM are 0.64 m/s and 4.45%, which are the optimal metrics, indicating that QRMGM has the highest prediction accuracy in the Dataset 2. By comparing the point prediction metrics, QRMGM still has the highest prediction accuracy in the Dataset 3 and 4, which indicates that the proposed method QRMGM is a competitive wind speed prediction method in terms of prediction accuracy.

③ Comparing the training time of the four neural network methods, in the Dataset 1, TT of QRMGM, QRLSTM, QRGRU and QRNN are 50.7 s, 139.7 s, 90.9 s and 55.3 s, respectively. Obviously, the training time of QRMGM is the least in the Dataset 1. By comparing the metric



(a) Interval prediction results of QRMGM in Dataset 1



(b) Interval prediction results of QRMGM in Dataset 2

Fig 5. Interval prediction results of QRMGM.

TT, the training time of QRMGM is still the shortest in Dataset 2–4. From the Dataset 1 to 4, the number of samples is gradually increasing, and the difference of training time between QRMGM and QRLSTM are 89.0 s, 94.1 s, 142.1 s and 185.2 s, respectively. It shows that as the number of training set samples increases, the difference in training time between the four models increases. The reduction in training time of QRMGM is attributed to the fact that the number of gates in gate structure unit network is reduced to one.

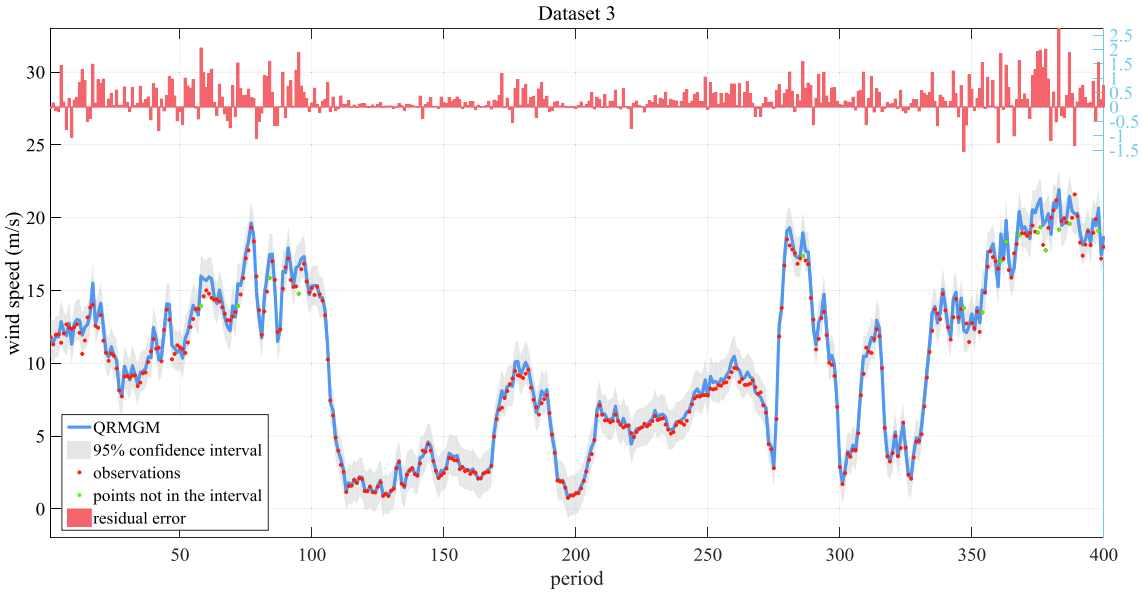
In summary, QRMGM significantly reduces training time without reducing prediction accuracy. The same conclusion can be obtained from the point prediction result comparison figures, as shown in Fig. 4. The two histograms are the ordering of the point prediction metrics.

(2) Interval prediction results evaluation

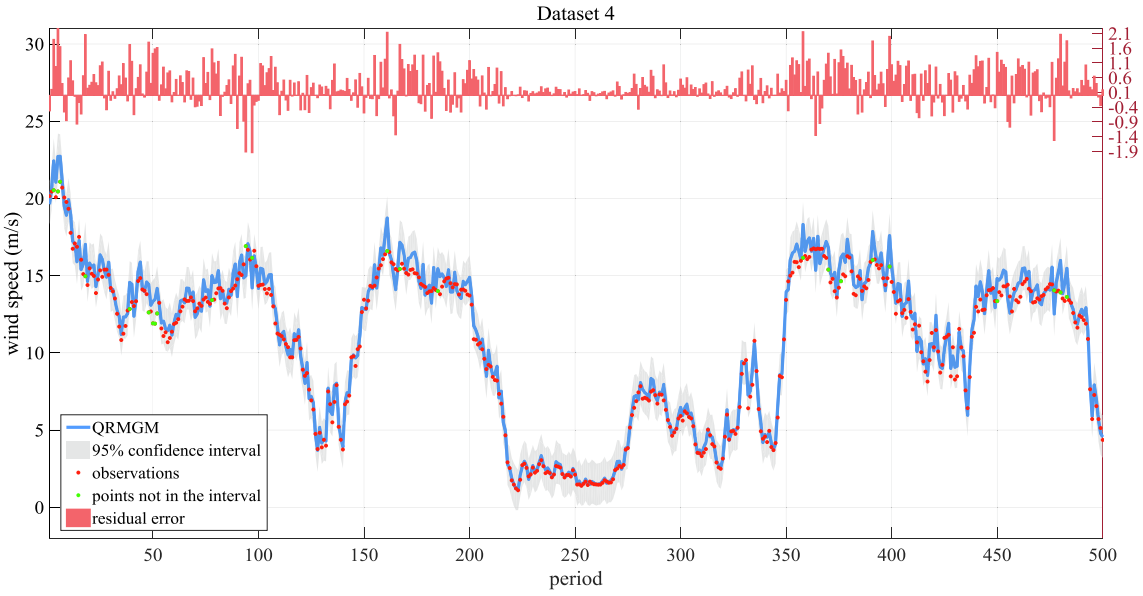
The interval prediction results evaluation is to verify the coverage probability and mean width of the interval, so as to judge whether the interval is suitable. The interval prediction evaluation metrics of seven methods on four datasets are shown in the Table 7. The best metrics in each dataset are highlighted with gray fill. The following three results can be analyzed from the Table 7:

① In the Dataset 1, the $CP_{95\%}$ of QRMGM, QRLSTM, QRGRU, QRNN, SVQR, GPR and QR are 0.952, 0.948, 0.943, 0.934, 0.955, 0.935 and 0.925, respectively. The coverage probability of seven methods on four datasets is close to 95%, indicating that the interval prediction of seven methods are reasonable.

② Taking the Dataset 1 as an example, the $MWP_{95\%}$ of QRMGM, QRLSTM, QRGRU, QRNN, SVQR, GPR and QR are 0.365, 0.365, 0.403, 0.569, 0.625, 0.569 and 0.739, respectively. Obviously, the prediction interval width of the three methods (QRMGM, QRLSTM, QRGRU) are significantly lower than those of the four methods (QRNN, SVQR, GPR, QR). Since the point prediction accuracy of the latter is lower than that of the former, the latter can only extend the interval width to make the CP close to 95%.



(c) Interval prediction results of QRMGM in Dataset 3



(d) Interval prediction results of QRMGM in Dataset 4.

Fig 5. (*continued*)

Table 8
Probability prediction evaluation metrics (CRPS).

model	Datasets	Dataset 1	Dataset 2	Dataset 3	Dataset 4
QRMGM	min	0.183	0.311	0.317	0.348
	mean	0.209	0.346	0.343	0.366
	max	0.224	0.372	0.361	0.401
QRLSTM	min	0.184	0.313	0.325	0.360
	mean	0.208	0.357	0.346	0.370
	max	0.231	0.403	0.374	0.378
QRGRU	min	0.226	0.389	0.411	0.417
	mean	0.241	0.422	0.424	0.446
	max	0.268	0.466	0.438	0.469
QRNN	min	0.334	0.568	0.546	0.610
	mean	0.351	0.624	0.604	0.637
	max	0.382	0.674	0.656	0.665
SVQR	mean	0.350	0.648	0.616	0.683
GPR	mean	0.366	0.568	0.586	0.614
QR	mean	0.491	0.755	0.852	0.852

③ In the Dataset 1, the $MC_{95\%}$ of QRMGM, QRLSTM, QRGRU, QRNN, SVQR, GPR and QR are 0.383, 0.385, 0.427, 0.609, 0.654, 0.608 and 0.798, respectively. The MC metric of QRMGM is optimal in the Dataset 1, showing that its prediction interval covers as many observation points as possible with as small width as possible. In general, the MC metric of QRLSTM is optimal in Dataset 2 and the MC metric of QRMGM is optimal in Dataset 1, 3 and 4.

In summary, the prediction interval obtained by QRMGM is most suitable. The interval prediction results of QRMGM on four datasets are plotted in Fig. 5. It can be seen from the figure that most of the observation points fall within the prediction interval and the interval width is narrow, which also shows that the prediction interval of QRMGM is very suitable.

(3) Probability prediction results evaluation

The probability prediction results evaluation is to verify comprehensive performance of probability prediction. The probability

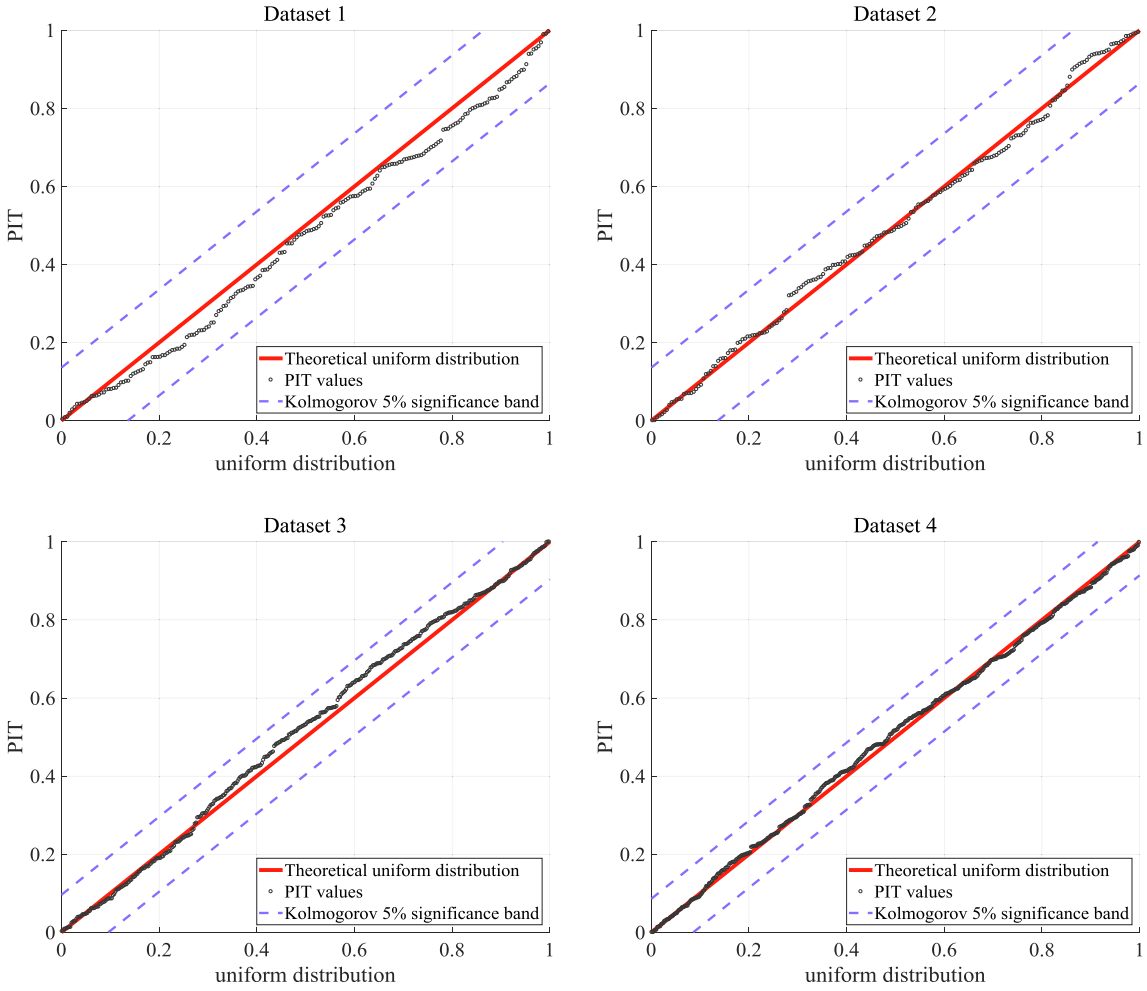


Fig 6. Reliability verification of QRMGM on four datasets.

prediction evaluation metrics of seven methods on four datasets are shown in the Table 8. The best metrics in each dataset are highlighted with gray fill. In the Dataset 1, the CRPS of QRMGM, QRLSTM, QRGRU, QRNN, SVQR, GPR and QR are 0.209, 0.208, 0.241, 0.351, 0.350, 0.366 and 0.491, respectively. The CRPS of QRLSTM is smallest in the Dataset 1, indicating that the probability prediction comprehensive performance of QRLSTM is the best in the Dataset 1. From the Dataset 2 to 4, the CRPS of QRMGM are 0.346, 0.343 and 0.366, respectively, which are the optimal probability prediction evaluation metrics. The results show that the comprehensive performance of QRMGM is the best in the Dataset 2–4. The results of the probability prediction are consistent with the point prediction results and interval prediction results.

4.5. Task III: Verify the reliability

The reliability of QRMGM is verified by calculating the PIT values of the observations and analyzing whether these PIT values obey the uniform distribution. QQ plot is used to visually analyze whether the PIT values are subject to uniform distribution. The uniform probability plot of PIT values of QRMGM is drawn in the Fig. 6. The PIT values of the four datasets are evenly distributed around the diagonal and its range evenly covers [0, 1]. All PIT points are located in the Kolmogorov 5% significance band [19], which indicates that predicted PDF are not excessively high or low, or excessively wide or narrow. Therefore, the prediction results obtained by QRMGM are reliable and convincing.

4.6. Task IV: Show probabilistic forecast results

Nine probability density curves sampled by equal spacing obtained by QRMGM in dataset 4 are drawn in the Fig. 7. The shape of the nine probability density curves is very full, and there are no cases where it is excessively high or low, wide or narrow, which indicate the probability density curves is suitable. In period 125, 250, 438, 500, the observation is almost at the center of the curve. In period 1, 63, 188, the observation is near the center of the curve. It shows that the prediction accuracy of these periods is very high. In period 313, 375, observation are far from the center. The prediction error of these periods is high. In probability prediction results of a validation set, some observations are close to the center and other observations are off-center, which just indicates that the probabilistic forecast is reliable. If all points are at the center or far from the center, this probabilistic forecast results may not be convincing.

5. Conclusions

Wind speed prediction and its potential uncertainty is important for the planning and utilization of wind energy, which can provide insightful information to balance the risk and economic benefit. This study first introduces the existing method linear QR. Through this method, a framework of hybrid method combining QR and other point prediction model is summarized. Based on this framework, QRLSTM and QRGRU are proposed for the QRNN's shortcoming that it cannot consider the timing information for time series regression. In order to minimize the training time of gated structure memory networks, a

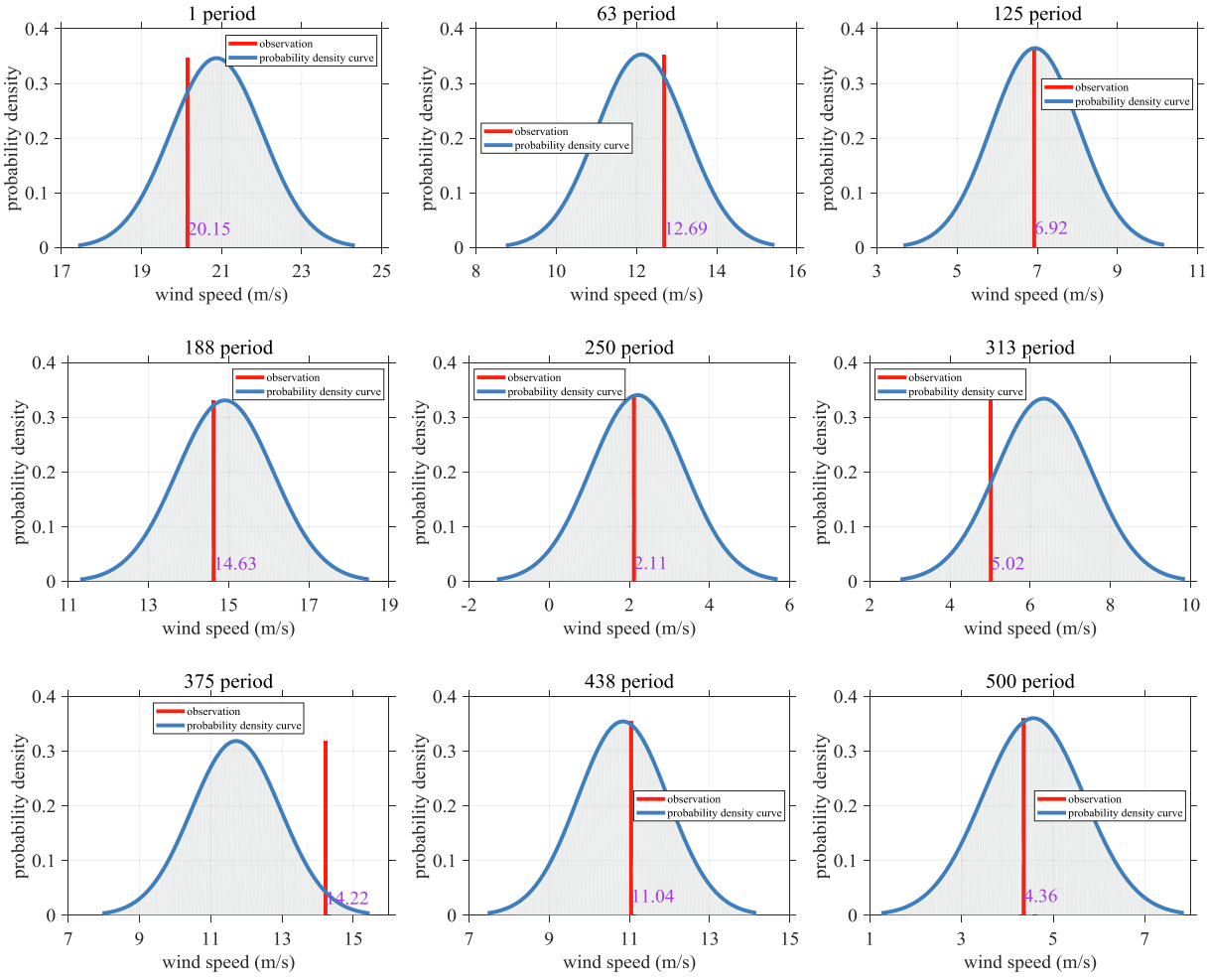


Fig 7. Probability density curves obtained by QRMGM in the dataset 4.

brand new model QRMGM is proposed. KDE is used to estimate the PDF of conditional quantiles obtained by QR, QRNN, QRLSTM, QRGRU and QRMGM. Grid search with cross-validation is used to select an appropriate bandwidth. The proposed methods are applied to solve four actual forecast case in Inner Mongolia, China. Six verification metrics R^2, CP_α, MWP_α, MC_α, CRPS, PIT and TT are used to evaluate point prediction accuracy, interval prediction suitability, probability prediction comprehensive performance, forecast reliability and training time.

The experimental results show that (1) The point prediction accuracy of QRMGM is the highest in the seven comparison models, and the time spent on training is also the least among the four neural networks, which is due to the design of the QRMGM network structure. (2) The prediction interval obtained by QRMGM is the most suitable, which covers as many observation points as possible with as small width as possible. (3) The comprehensive performance of the probability prediction results obtained by QRMGM is also optimal, and the prediction result is also reliable through the reliability test.

In summary, QRMGM obtains high-precision point prediction, appropriate prediction interval and high-performance probabilistic prediction results with minimal training time. These results fully demonstrate that QRMGM has the ability to obtain wind speed prediction results with excellent performance on accuracy, uncertainty and reliability.

Declaration of Competing Interest

None declared.

Acknowledgements

This work is supported by the National Key R&D Program of China (2017YFC0405900), the National Natural Science Foundation of China (No. 91647114, 51809098, 61703199), the National Public Research Institutes for Basic R & D Operating Expenses Special Project (CKSF2017061/SZ), and special thanks are given to the anonymous reviewers and editors for their constructive comments.

References

[1] Zhang Z, Qin H, Liu Y, Wang Y, Yao L, Li Q, et al. Long Short-Term Memory Network based on Neighborhood Gates for processing complex causality in wind speed prediction. Energy Convers Manage 2019;192:37–51.
[2] Liang Z, Liang J, Wang C, Dong X, Miao X. Short-term wind power combined forecasting based on error forecast correction. Energy Convers Manage 2016;119:215–26.
[3] Foley AM, Leahy PG, Marvuglia A, McKeogh EJ. Current methods and advances in forecasting of wind power generation. Renew Energy 2012;37:1–8.
[4] Andrade JR, Bessa RJ. Improving renewable energy forecasting with a grid of numerical weather predictions. IEEE T Sustain Energy 2017;8:1571–80.
[5] Al-Yahyai S, Charabi Y, Gastli A. Review of the use of Numerical Weather Prediction (NWP) Models for wind energy assessment. Renew Sustain Energy Rev 2010;14:3192–8.
[6] Zhang C, Zhou J, Li C, Fu W, Peng T. A compound structure of ELM based on feature selection and parameter optimization using hybrid backtracking search algorithm for wind speed forecasting. Energy Convers Manage 2017;143:360–76.
[7] Erdem E, Shi J. ARMA based approaches for forecasting the tuple of wind speed and direction. Appl Energy 2011;88:1405–14.
[8] Chen K, Yu J. Short-term wind speed prediction using an unscented Kalman filter based state-space support vector regression approach. Appl Energy 2014;113:690–705.
[9] Li G, Shi J. On comparing three artificial neural networks for wind speed

forecasting. Appl Energy 2010;87:2313–20.
[10] Peng T, Zhou J, Zhang C, Zheng Y. Multi-step ahead wind speed forecasting using a hybrid model based on two-stage decomposition technique and AdaBoost-extreme learning machine. Energy Convers Manage 2017;153:589–602.
[11] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44.
[12] Shi Z, Liang H, Dinavahi V. Direct interval forecast of uncertain wind power based on recurrent neural networks. IEEE T Sustain Energy 2018;9:1177–87.
[13] Ku K, Mak MW, Sin WC. A study of the lamarckian evolution of recurrent neural networks. IEEE T Evolut Comput 2000;4:31–42.
[14] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735–80.
[15] Qin Y, Li K, Liang Z, Lee B, Zhang F, Zhang L, et al. Hybrid forecasting model based on long short term memory network and deep learning neural network for wind signal. Appl Energy 2019;236:262–72.
[16] Gers FA, Schmidhuber J. Recurrent nets that time and count. IEEE 2000:189–94.
[17] Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, et al. Learning phrase representations using RNN encoder-decoder for statistical. Mach Transl 2014.
[18] Greff K, Srivastava RK, Koutnik J, Steunebrink BR, Schmidhuber J. LSTM: a search space odyssey. IEEE T Neur Net Lear 2017;28:2222–32.
[19] Liu Y, Ye L, Qin H, Hong X, Ye J, Yin X. Monthly streamflow forecasting based on hidden Markov model and Gaussian Mixture Regression. J Hydrol 2018;561:146–59.
[20] Zhang Z, Ye L, Qin H, Liu Q, Wang C, Yu X, et al. Wind speed prediction method using shared weight long short-term memory network and gaussian process regression. Appl Energy 2019;247:270–84.
[21] Men Z, Yee E, Lien F, Wen D, Chen Y. Short-term wind speed and power forecasting using an ensemble of mixture density neural networks. Renew Energy 2016;87:203–11.
[22] Nielsen HA, Madsen H, Nielsen TS. Using quantile regression to extend an existing wind power forecasting system with probabilistic forecasts. Wind Energy 2006;9:95–108.
[23] Koenker R, Hallock KF. Quantile regression. J Econ Perspect 2001;15:143–56.
[24] Cannon AJ. Quantile regression neural networks: Implementation in R and application to precipitation downscaling. Comput Geosci-Uk 2011;37:1277–84.
[25] He Y, Li H. Probability density forecasting of wind power using quantile regression neural network and kernel density estimation. Energy Convers Manage 2018;164:374–84.
[26] Wang GB, Wang HZ, Li GQ, Peng JC, Liu YT. Deep belief network based deterministic and probabilistic wind speed forecasting approach. Appl Energy 2016;182:80–93.
[27] Epanechnikov VA. Non-parametric estimation of a multivariate probability density. Theor Probab Appl+ 1969;14:153–8.
[28] Guo SW, Lin DY. Regression analysis of multivariate grouped survival data. Biometrics 1994;50:632–9.
[29] Zhao J, Zurada JM, Yang J, Wu W. The convergence analysis of SpikeProp algorithm with smoothing L-1/2 regularization. Neural Netw 2018;103:19–28.
[30] Tenreiro C. Fourier series-based direct plug-in bandwidth selectors for kernel density estimation. J Nonparametr Stat 2011;23:533–45.
[31] Duong T, Hazelton ML. Cross-validation bandwidth matrices for multivariate kernel density estimation. Scand J Stat 2005;32:485–506.
[32] Reshef D, Reshef Y, Mitzenmacher M, Sabeti P. Equitability Analysis of the Maximal Information Coefficient, with Comparisons. 2013.
[33] Guo Z, Uhrig RE. Using genetic algorithms to select inputs for neural networks. IEEE Comput. Soc. Press; 1992. p. 223–34.
[34] Li R, Jin Y. A wind speed interval prediction system based on multi-objective optimization for machine learning method. Appl Energy 2018;228:2207–20.
[35] Liu Y, Ye L, Qin H, Ouyang S, Zhang Z, Zhou J. Middle and long-term runoff probabilistic forecasting based on Gaussian mixture regression. Water Resour Manage 2019;33(5):1785–99.
[36] Laio F, Tamea S. Verification tools for probabilistic forecasts of continuous hydrological variables. Hydrol Earth Syst Sci 2007;11:1267–77.
[37] Deep Tang Y. Learning using linear support vector machines. Comput Sci 2013.
[38] Jain P, Netrapalli P, Kakade SM, Kidambi R, Sidford A. Parallelizing stochastic gradient descent for least squares regression: mini-batching, averaging, and model misspecification. J Mach Learn Res 2018;18.
[39] Lee H, Kim N, Lee J. Deep neural network self-training based on unsupervised learning and dropout. Int J Fuzzy Logic Intell Syst 2017;17:1–9.
[40] Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. 2014.