water

MDPI

Article

A Novel Daily Runoff Probability Density Prediction Model Based on Simplified Minimal Gated Memory–Non-Crossing Quantile Regression and Kernel Density Estimation

Huaiyuan Liu [1,2,3], Sipeng Zhu [1,2,3] and Li Mo [1,2,3,*]

[1] School of Civil and Hydraulic Engineering, Huazhong University of Science and Technology, Wuhan 430074, China; huaiyuanliu@hust.edu.cn (H.L.); zsp960512@163.com (S.Z.)
[2] Provincial Key Laboratory of Digital Watershed Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China
[3] Institute of Water Resources and Hydropower, Huazhong University of Science and Technology, Wuhan 430074, China
* Correspondence: moli@hust.edu.cn

Abstract: Reliable and accurate daily runoff predictions are critical to water resource management and planning. Probability density predictions of daily runoff can provide decision-makers with comprehensive information by quantifying the uncertainty of forecasting. Models based on quantile regression (QR) have been proven to achieve good probabilistic prediction performance, but the predicted quantiles may crossover with each other, seriously reducing the reliability of the prediction. This paper proposes non-crossing quantile regression (NCQR), which guarantees that the intervals between adjacent quantiles are greater than 0, which avoids the occurrence of quantile crossing. In order to apply NCQR to the prediction of nonlinear runoff series, this paper combines NCQR with recurrent neural network (RNN) models. In order to reduce the model training time and further improve the model accuracy, this paper simplifies the minimal gated memory (MGM) model and proposes a new RNN model, called the simplified minimal gated memory (SMGM) model. Kernel density estimation (KDE) is used to transform the discrete quantiles predicted using SMGM-NCQR into a continuous probability density function (PDF). This paper proposes a novel daily density prediction model that combines SMGM-NCQR and KDE. Three daily runoff datasets in the Yangtze River Basin in China are taken as examples and compared with the advanced models in current research in terms of five aspects: point prediction evaluation, interval prediction evaluation, probability density prediction evaluation, the degree of quantile crossing and training time. The experimental results show that the model can provide high-quality and highly reliable runoff probability density predictions.

Keywords: daily runoff prediction; probability density prediction; non-crossing quantile regression; simplified minimal gated memory network; kernel density estimation

Citation: Liu, H.; Zhu, S.; Mo, L. A Novel Daily Runoff Probability Density Prediction Model Based on Simplified Minimal Gated Memory–Non-Crossing Quantile Regression and Kernel Density Estimation. *Water* **2023**, *15*, 3947. https://doi.org/10.3390/w15223947

Academic Editor: Haixing Liu

Received: 25 September 2023
Revised: 4 November 2023
Accepted: 7 November 2023
Published: 13 November 2023

1. Introduction

Daily runoff prediction has played a significant role in water resource management and planning [1,2]. Due to the comprehensive influence of various factors, such as geological conditions, human activities and climate change in the watershed, daily runoff series often have highly nonlinear and non-stationary characteristics [3,4], which introduce difficulties to daily runoff prediction. Finding accurate and reliable daily runoff models is currently an important and popular research direction in the field of hydrology [5,6].

The current models for predicting daily runoff can be divided into process-driven models and data-driven models [7,8]. Process-driven models, such as MIKESHE [9] and SWAT [10], have strong physical interpretability, but the large number of computational resources required and complex modeling processes limit their practical use. The data-driven model directly constructs linear and nonlinear relationships between prediction factors

and the predicted runoff. Data-driven models can be divided into statistical models, machine learning models and deep learning models. Statistical models such as autoregressive integrated moving average (ARIMA) [11] and seasonal autoregressive integrated moving average (SARIMA) [12] use statistical methods to predict runoff. Most statistical models are linear models based on stationarity assumptions, which cannot accurately predict nonlinear and non-stationary runoff sequences. Compared with statistical models, machine learning models such as support vector regression (SVR) [13] and the adaptive-network-based fuzzy inference system (ANFIS) [14] can describe the nonlinearity of runoff sequences. However, the simple structure of machine learning models limits their ability to extract information, which limits the performance of the model. Deep learning models, represented by recurrent neural network (RNN) models, are widely used in runoff forecasting tasks due to their ability to effectively capture nonlinearity and time dependence in runoff sequences [15]. When the sequence of model inputs is too long, the RNN model may encounter the problem of gradient explosion or gradient disappearance [16]. To address this issue, Hochreiter et al. [17] introduced gating memory units in traditional RNN models and proposed the long short-term memory (LSTM) model. There are three gates in the model structure of LSTM: the input gate, forgetting gate and output gate. Rahimzad et al. [18] applied LSTM to daily runoff prediction tasks and achieved excellent results.

The gated recurrent units (GRUs) model is an improved version of the LSTM model proposed by Cho et al. [19]. The GRU model only has two gates, namely the reset gate and the update gate. Compared to the LSTM model, the GRU model has fewer parameters and a shorter training time [20]. The minimal gated memory (MGM) [21,22] model is another simplified version of LSTM, which combines the input gate and forgetting gate of the LSTM model and removes the output gate, so the MGM model only has one gate. Compared with the LSTM model and GRU model, the MGM model has a simpler structure and fewer parameters. An RNN model with fewer parameters and higher accuracy is currently one of the research objectives for improving RNN models. Therefore, this article proposes a simplified version of the MGM called the simplified minimal gated memory (SMGM) model.

The above models are all point prediction models that only provide predicted values close to the observed values without considering the error between the observed values and predicted values. When the prediction results with significant errors are applied to flood control and water resource planning, it will create safety hazards and reduce the efficiency of water resource utilization. The probabilistic prediction model of runoff has attracted widespread attention from researchers in the hydrological field because it can quantify the uncertainty of prediction and provide more effective information to decision-makers compared to point prediction. Generally, probabilistic runoff prediction methods in current studies can be divided into two categories: parametric models and non-parametric models [23].

Parametric models such as GPR [24,25] and DeepAR [26] assume that runoff conforms to a certain distribution and that only a few parameters in the distribution need to be estimated. However, the assumed distribution of parameter models may significantly differ from the actual distribution of daily runoff, so satisfactory predictive performance may not be obtained. Non-parametric models do not make any assumptions but study the data distribution based on the characteristics of the data itself. Quantile regression (QR) [27–29] is a classic non-parametric probability prediction method. By predicting a set of quantiles, multiple prediction intervals at different confidence levels can be obtained, which quantifies the uncertainty of forecasting. However, QR is a linear model [30] and cannot solve nonlinear problems. The model proposed by Jahangir et al. [31] that combines LSTM and QR can learn the nonlinearity of runoff while quantifying uncertainty. For QR-based models, a single model can only predict a single quantile. But when multiple prediction intervals with different confidence levels and probability density functions need to be obtained, a large number of quantiles need to be predicted. This requires training a large number of QR-based models, which requires a significant amount of computing time

and computational memory. Wang et al. [32] improved QR and increased the number of QR outputs from 1 to the number of required quantiles. The improved QR increases the number of predicted quantiles without reducing prediction accuracy. However, because all quantiles are predicted at the same time without relevant constraints in the model based on QR, the predicted quantiles may cross. For example, assuming $0 < \tau_1 < \tau_2 < 1$, the τ_1th quantile is greater than the τ_2th quantile. Quantile crossing seriously affects the reliability and prediction accuracy of the predicted quantiles. Therefore, in order to improve the reliability of QR, this paper proposes a new non-crossing quantile regression (NCQR) approach that can avoid the occurrence of quantile crossing.

The output of the NCQR-based model is a series of discrete quantiles, which can provide prediction intervals with different confidence levels, but the probability density function (PDF) can provide a more comprehensive, intuitive and detailed prediction compared to interval prediction. Kernel density estimation (KDE) [33] is a classic non-parametric estimation method that does not require any assumptions about the shape of the distribution and converts discrete quantiles into PDFs. He et al. [34] obtained PDFs by using the quantiles predicted by a combination model based on an RNN and QR as an input for KDE. QR-based models are often accompanied by quantile crossing, and non-monotonically increasing quantiles can lead to ineffective PDF prediction through KDE, which impedes further analysis. The NCQR-based model proposed in this article can generate non-crossing quantiles, and inputting these quantiles into KDE can obtain reliable PDFs.

This study proposes a new model for daily runoff probability density prediction combining SMGM, NCQR and KDE. The combined model of SMGM and NCQR predicts reliable quantile sets, and KDE is used to estimate the continuous PDF. The main contributions of this article are as follows:

(1) This article proposes NCQR, which can avoid quantile crossing in predicted quantiles.
(2) In order to reduce training time and improve model accuracy, this article simplifies the model structure of MGM and proposes SMGM.
(3) The combination model based on SMGM and NCQR can efficiently generate reliable quantiles, which KDE converts into continuous PDFs.
(4) All models in this article are compared from multiple perspectives using multiple evaluation metrics in three daily runoff datasets. The experimental results show that the model can efficiently obtain reliable and accurate probability density prediction results.

The remainder of this article is organized as follows: Section 2 introduces the model proposed in this article. Section 3 introduces the evaluation metrics used in this article. Section 4 provides a case study using three datasets to evaluate the model proposed in this article. Section 5 provides the conclusion of this article.

2. Methods

2.1. Simplified Minimal Gated Memory Network

MGM is a simplified version of LSTM. MGM combines the forgetting gate and input gate of LSTM into one gate and removes the output gate of LSTM, so MGM only has one gate. Compared to LSTM with three gates and GRU with two gate structures, MGM has a simpler structure. The model structure of MGM is shown in (1)–(3):

$$\mathbf{f}_t = \sigma\left(W_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f\right) \tag{1}$$

$$\widetilde{\mathbf{h}}_t = \tanh\left(W_h\left[\mathbf{f}_t \odot \mathbf{h}_{t-1}, \mathbf{x}_t\right] + \mathbf{b}_h\right) \tag{2}$$

$$\mathbf{h}_t = (1 - \mathbf{f}_t) \odot \mathbf{h}_{t-1} + \mathbf{f}_t \odot \widetilde{\mathbf{h}}_t \tag{3}$$

where W_f and W_h are the weight parameters, $\mathbf{b}_f$ and $\mathbf{b}_h$ are the bias parameters, $\mathbf{x}_t$ is the model input at time t, $\mathbf{h}_t$ is the cell state at time t, and tanh and σ represent the tanh activation function and sigmoid activation function, respectively. And $\odot$ represents the Hadamard product.

$\mathbf{f}_t$, as the only gate of MGM, controls the mapping of model inputs to model outputs. In order to learn the relationship between the model input and the target, a training set is used to train the model with the goal of minimizing the loss function. To more effectively learn the relationships needed, the form of $\mathbf{f}_t$ can be optimized. Therefore, this article further simplifies the gate of MGM and proposes SMGM. The SMGM gate equation removes the cell state $\mathbf{h}_t$ at time t and only saves the model input $\mathbf{x}_t$ at time t and bias $\mathbf{b}_f$. The gate equation of SMGM is Equation (4). And the other parts of the SMGM model are the same as those of MGM. The model structure of SMGM is shown in Figure 1. Meanwhile, this article proposes the SMGM-NCQR model for predicting reliable quantile sets of nonlinear daily runoff series.

$$\mathbf{f}_t = \sigma\left(W_f \mathbf{x}_t + \mathbf{b}_f\right) \tag{4}$$

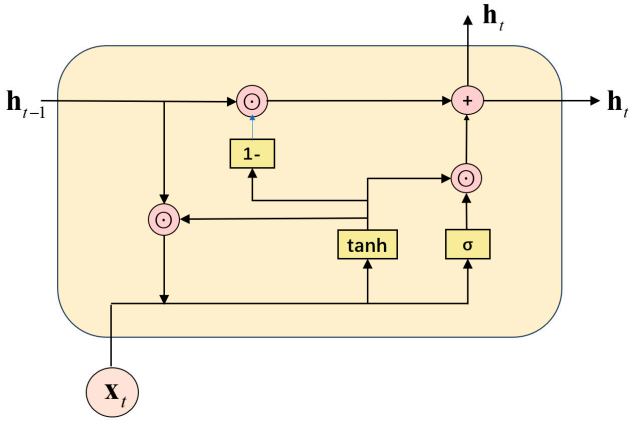

Figure 1. The model structure of SMGM.

2.2. Novel Non-Crossing Quantile Regression

Traditional regression can only output the mean or median of future daily runoff. Quantile regression (QR) can predict quantiles of daily future runoff, which can quantify the uncertainty of prediction. The definition of QR is as follows:

$$Q_{y_i}(\tau \mid x_i) = \beta(\tau) x_i \quad i = 1, 2, \cdots, T \tag{5}$$

where $Q_{y_i}(\tau \mid x_i)$ is the τth quantile of the dependent variable y_i under the independent variable x_i and $\tau \in (0, 1)$, $\beta(\tau)$ is the regression coefficient vector and the loss function of QR is

$$Loss(\tau) = \sum_{i=1}^{T} \varphi_\tau(y_i - x_i \beta(\tau)) \tag{6}$$

$$\varphi_\tau(u) = \begin{cases} \tau u & u \geq 0 \\ (\tau - 1)u & u < 0 \end{cases} \tag{7}$$

However, for traditional QR, a model can only generate a single quantile. In order to obtain the probability density function and multiple prediction intervals with different confidence levels, a large number of quantiles need to be predicted, which requires the construction and training of a large number of QR models. Wang et al. [32] increased the number of QR outputs without loss of accuracy. The improved QR can predict all required quantiles $Q_{y_i}(\tau \mid x_i) = \left\{Q_{y_i}(\tau_q \mid x_i)\right\}_{q=1,2,\cdots,Q}$ with one model. The loss function of the improved QR is

$$L = \sum_{q=1}^{Q} \text{Loss}\left(\tau_q\right) = \frac{1}{Q} \sum_{q=1}^{Q} \sum_{i=1}^{T} \varphi_{\tau_q}\left(y_i - x_i \beta(\tau_q)\right) \tag{8}$$

However, these quantiles are simultaneously generated without constraints, and quantile crossing usually occurs. In order to avoid the above problem, this paper proposes a new non-crossing quantile regression (NCQR) approach.

NCQR reconstructs the outputs of QR into three parts: the first quantile $Q_{y_i}(\tau_1 \mid x_i)$, the difference L between the last quantile $Q_{y_i}(\tau_Q \mid x_i)$ and the first quantile $Q_{y_i}(\tau_1 \mid x_i)$, and the ratio set $\{\theta_n\}_{n=1,2,\cdots,Q-1}$. θ_n is the ratio of the interval between $Q_{y_i}(\tau_{n+1} \mid x_i)$ and $Q_{y_i}(\tau_n \mid x_i)$ to L. Then, the estimated values of all quantiles can be obtained using the following formula:

$$Q_{y_i}(\tau_m \mid x_i) = Q_{y_i}(\tau_1 \mid x_i) + \sum_{j=1}^{m-1} \theta_J l \quad m = 1, 2, \cdots, Q \tag{9}$$

In order to obtain valid and reliable quantiles, the outputs of NCQR need to meet the following constraints.

$$L = Q_{y_i}(\tau_Q \mid x_i) - Q_{y_i}(\tau_1 \mid x_i) > 0 \tag{10}$$

$$\theta_n = \frac{Q_{y_i}(\tau_{n+1} \mid x_i) - Q_{y_i}(\tau_n \mid x_i)}{L} > 0 \quad n = 1, 2, \cdots, Q-1 \tag{11}$$

$$\sum_{n=1}^{Q-1} \theta_n = 1 \tag{12}$$

The purpose of this restriction is to avoid the occurrence of quantile crossing by ensuring that the distance between adjacent quantiles is greater than 0 in order to meet the limiting conditions specified in Equations (10)–(12). In the output layer, the neuron of output L is activated by the Softplus function, ensuring that the output result is greater than 0, which satisfies constraint (10). The neurons of output $\{\theta_n\}_{n=1,2,\cdots,Q-1}$ are activated by the Softmax function, ensuring that their outputs are greater than 0 and the sum is 1, which satisfies constraints (11) and (12). Figure 2 shows the model structure of NCQR. The definitions of the Softmax function and Softplus function are as follows:

$$\text{Softmax}(o_i) = \frac{e^{o_i}}{\sum_{i=1}^{N} e^{o_i}} \; i = 1, 2, \cdots N \tag{13}$$

$$\text{Softplus}(o_i) = \log(1 + e^{o_i}) \tag{14}$$

2.3. Kernel Density Estimation

This study used KDE to convert the discrete quantiles predicted using SMGM-NCQR into continuous PDFs. KDE is a classic non-parametric estimation method that estimates the PDF of random variables from a set of quantiles without assuming the distribution of data beforehand. Unlike parametric estimation methods, which can only predict parameters of parametric distributions such as Gaussian distribution and further obtain fixed-form PDFs, KDE can output any form of PDFs. The definition of KDE is as follows:

$$PDF(y_i) = \frac{1}{QB} \sum_{q=1}^{Q} K\left(\frac{Q_{y_i}(\tau_q \mid x_i) - y_i}{B}\right) \tag{15}$$

$$K(\alpha) = \begin{cases} \frac{3}{4}(1 - \alpha^2) & \alpha \in [-1, 1] \\ 0 & \alpha \notin [-1, 1] \end{cases} \tag{16}$$

where Q is the number of samples, B is the bandwidth, $PDF(y_i)$ is the PDF of y_i. K is a kernel function. The Epanechnikov kernel function was chosen for this study because it can achieve the smallest root-mean-square deviation compared to other kernel functions [35].

The formula for Epanechnikov kernel function is shown in Equation (16). Bandwidth is a key parameter that has a significant impact on KDE performance. Choosing a bandwidth that is too large or too small can result in significant errors in the KDE results. In this study, a grid search method based on cross-validation was used to determine the bandwidth. Figure 2 below shows the probability density prediction model based on SMGM-NCQR and KDE.

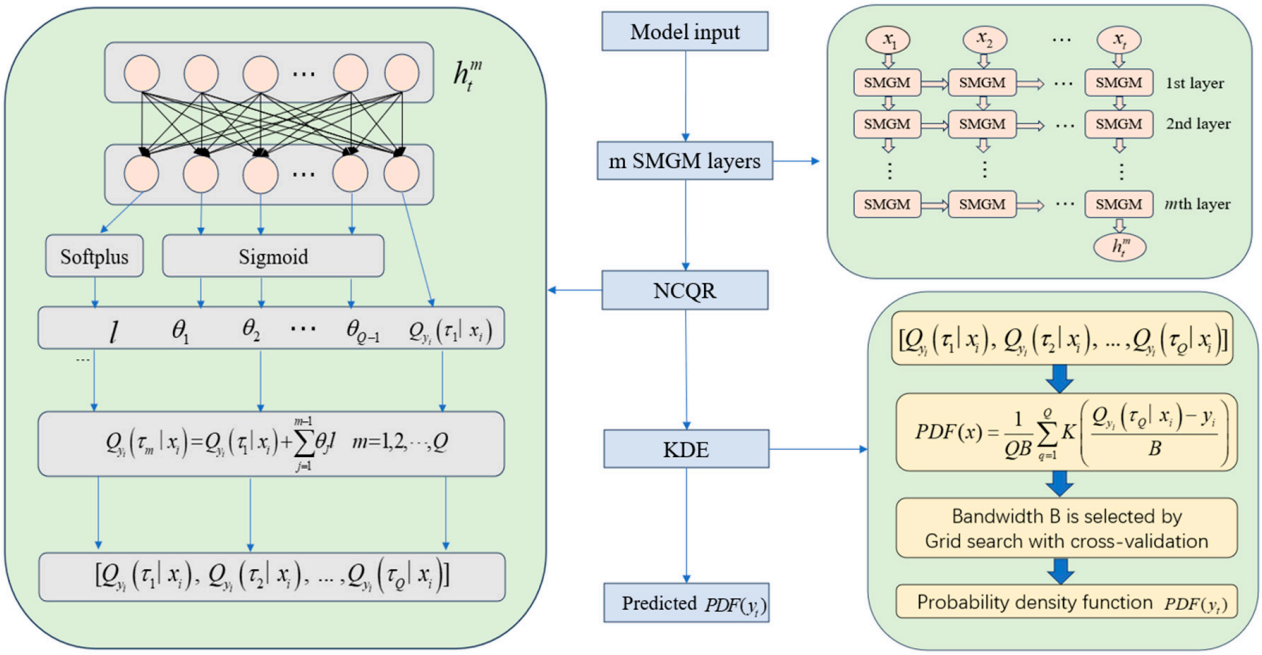

Figure 2. The probability density prediction model based on SMGM-NCQR and KDE.

2.4. Maximal Information Coefficient

The MIC method was proposed by Reshef et al. [36] to quantify the correlation between two variables. The advantage of the MIC method is that it cannot only identify specific functional relationships, such as linear, nonlinear and sine functions, but also identify complex relationships that are difficult to represent using functions, with strong universality. The range of values for the MIC is [0, 1]. The closer the value of the MIC is to 1, the stronger the correlation between the two variables. This study used the MIC to calculate the correlation between current runoff and historical runoff for selecting input features for the model. The definition of the MIC is as follows:

$$I(X;Y) = \int p(X,Y)\log_2 \frac{p(X,Y)}{p(X)p(Y)} dXdY \tag{17}$$

$$\mathrm{MIC}(X;Y) = \max_{a^*b<B(D)} \frac{I(X;Y)}{\log_2 \min(a,b)} \tag{18}$$

$$\mathrm{B}(|\mathrm{D}|) = |\mathrm{D}|^{0.6} \tag{19}$$

where X and Y are two variables; $p(X,Y)$ is the joint probability density of A and B; $p(X)$ and $p(Y)$ are the edge probability density of X and Y, respectively; a and b are the number of grids divided on the X and Y axes; and D is the amount of data.

2.5. Framework of the Proposed Combined Model

The framework of the proposed composite model is shown in Figure 3. The model mainly consists of four main steps. Each step is described as follows:

Step 1: The MIC was used to calculate the correlation between historical runoff and current runoff, and historical runoff with an MIC value greater than 0.85 was selected as

the input feature of the model. The dataset was normalized and divided into the training set and test set. The normalization formula is as follows:

$$x_{\text{normal}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{20}$$

where x_{normal} is the normalized runoff sequence; x is the original runoff sequence; and $x_{\max}$ and $x_{\min}$ are the maximum and minimum values, respectively.

Step 2: The MGM-NCQR model was built and trained on the training set with the goal of minimizing the loss function, which is Formula (4).

Step 3: The test set was input into the trained MGM-NCQR model, and anti-normalization was carried out on the obtained conditional quantile set, which was input into KDE to obtain the predicted probability density prediction results. The conditional quantile of $\tau = 0.5$ obtained in the continuous probability density curve is the result of point prediction. The quantiles of $\tau_1 = (1-\alpha)/2$ and $\tau_2 = (1+\alpha)/2$ are the lower and upper boundaries of the interval prediction result with a confidence level of α, respectively. The formula for anti-normalization is as follows:

$$Q_{y_t}(\tau \mid x_t) = Q_{y_t,\text{normal}}(\tau \mid x_t) \bullet (x_{\max} - x_{\min}) + x_{\min} \tag{21}$$

where $Q_{y_t}(\tau \mid x_t)$ is the quantile after anti-normalization.

Step 4: The performance of the model was evaluated using different evaluation metrics.

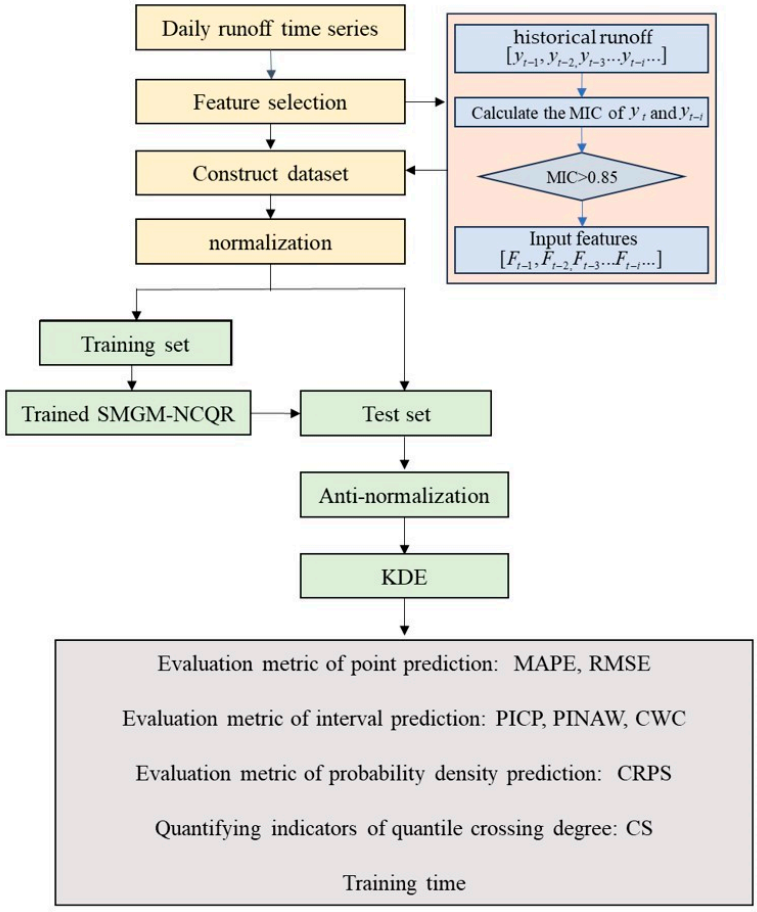

Figure 3. Framework of the proposed hybrid model.

3. Model Evaluation Metrics

In order to comprehensively evaluate the performance of the proposed runoff probability density prediction model, this study used the point prediction evaluation metric, interval

prediction evaluation metric, degree of quantile crossing, probability density prediction evaluation metric and training time to compare with other models. They are introduced in the following subsections.

3.1. Evaluation Metric of Point Prediction

This study used the root-mean-square deviation (RMSE) and mean absolute percentage error (MAPE) to evaluate the point prediction performance. RMSE and MAPE reflect the degree of difference between predicted runoff series and observed runoff series. The smaller the value of RMSE and MAPE, the higher the accuracy of point prediction. The formulas of RMSE and MAPE are as follows, respectively:

$$MAPE = \frac{1}{T}\sum_{t=1}^{T}\frac{\left|y' - y_t^{obs}\right|}{y_t^{obs}} \tag{22}$$

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(y_t^{obs} - y_t^{p}\right)^2} \tag{23}$$

where y_t^{obs} is the observed value at time t, y_t^p is the predicted value at time t, T is the length of the predicted runoff sequence and $\overline{y}^{obs}$ is the average value of the observed runoff sequence.

3.2. Evaluation Metric of Interval Prediction

In order to evaluate the interval prediction performance of models, the prediction interval coverage probability (PICP), the prediction interval normalized average width (PINAW) and the coverage-width-based standard (CWC) were used in this study. The PICP represents the probability that the observed value will eventually fall into the predicted interval. The larger the PICP, the more values there are within the predicted range. The valid interval prediction of PICPα should be greater than the preset confidence level α; otherwise, it is an invalid prediction, and PICPα is defined as follows:

$$\mathrm{PICP}_\alpha = \frac{1}{T}\sum_{t=1}^{T}\gamma_t^\alpha \tag{24}$$

$$\gamma_t^\alpha = \begin{cases} 1, & y_t \in [L_t^\alpha, U_t^\alpha] \\ 0, & y_t \notin [L_t^\alpha, U_t^\alpha] \end{cases} \tag{25}$$

where L_t^α and U_t^α are the lower and upper bounds of the predicted interval, respectively, with the preset confidence level α at time t.

PINAW is another important metric for evaluating interval predictions. It is used to measure the proportion of the width of the prediction interval to the difference between the maximum and minimum values of the observed runoff sequence. The formula for PINAW is as follows:

$$\mathrm{PINAW}_\alpha = \frac{1}{T}\sum_{t=1}^{T}\frac{U_t^\alpha, -L_t^\alpha,}{D} \tag{26}$$

where D is the difference between the maximum value and the minimum value of the observed runoff sequence.

Using PICP as the only evaluation metric for interval prediction is not reasonable, as it can be improved by increasing the width of the predicted interval. An excessively wide prediction interval cannot convey effective uncertain information to decision-makers. The ideal interval prediction model should achieve a small PINAWα when PICPα is larger than α. Under the premise of effective interval prediction, the narrower the predicted interval, the more information it can provide. PINAWα and PICPα are not independent of each other, and an increase in PINAW may cause an increase in PICPα and vice versa. Therefore,

the coverage-width-based criterion (CWCα) that considers both PICPα and PINAWα was selected as a comprehensive measure for interval prediction. The smaller the CWCα of interval prediction, the better the interval prediction performance. The formula of CWCα is as follows:

$$\mathrm{CWC}_\alpha = \mathrm{PINAW}_\alpha + \xi(\mathrm{PICP}_\alpha)e^{-\mu(\mathrm{PICP}_\alpha - \alpha)} \tag{27}$$

$$\xi(\mathrm{PICP}_\alpha) = \begin{cases} 0, \mathrm{PICP}_\alpha \geq \alpha \\ 1, \ \mathrm{PICP}_\alpha < \alpha \end{cases} \tag{28}$$

where μ is the scaling factor that amplifies the difference between PICPα and α when the PICP is less than the confidence level of α; otherwise, the value of CWC is equal to the value of PINAW.

3.3. Quantifying Indicators of Quantile Crossing Degree

This study used the constraint score (CS) to quantify the degree of quantile crossing. If the value of CS is equal to 0, it indicates that the quantiles predicted by the model have not undergone quantile crossing. If the value of CS is greater than 0, it indicates that the quantiles predicted by the model have undergone quantile crossing. The larger the CS, the more serious the crossing problem in quantile prediction. The definition of CS is as follows:

$$\mathrm{CS} = \sqrt{\frac{2\widetilde{\tau}}{T}\sum_{t=1}^{T}\sum_{m=1}^{M-1}\alpha_{t,m}^2} \tag{29}$$

$$\alpha_{t,m} = \begin{cases} 0, \hat{P}_t(\tau_m) \leq \hat{P}_t(\tau_{m+1}) \\ \hat{P}_t(\tau_m) - \hat{P}_t(\tau_{m+1}), \hat{P}_t(\tau_m) > \hat{P}_t(\tau_{m+1}) \end{cases} \tag{30}$$

$$\widetilde{\tau} = \tau_{m+1} - \tau_m \tag{31}$$

where $\hat{P}_t(\tau_m)$ is the predicted value of the mth quantile at time t.

3.4. Evaluation Metric of Probability Density Prediction

The continuous ranking probability score (CRPS) was used as the probability density prediction evaluation metric in this study. The smaller the CRPS, the better the probability density prediction performance. The definition of CRPS is as follows:

$$\mathrm{CRPS} = \frac{1}{T}\sum_{t=1}^{T}\int_{-\infty}^{+\infty}\left[F\left(y_t^p\right) - H\left(y_t^p - y_t^{obs}\right)\right]^2 dy_t \tag{32}$$

$$F\left(y_t^P\right) = \int_{-\infty}^{y_t^P} p(x)dx \tag{33}$$

$$H\left(y_t^p - y_t^{obs}\right) = \begin{cases} 0 & y_t^p < y_t^{obs} \\ 1 & y_t^p \geq y_t^{obs} \end{cases} \tag{34}$$

where $p(x)$ is the probability density function of x, and $F\left(y_t^P\right)$ is the cumulative distribution function of y_t.

4. Case Study

4.1. Study Area and Data

The upstream region of the Yangtze River was taken as the study area. The upper reach of the Yangtze River is the reach from the source of the Yangtze River to Yichang, Hubei Province, China. The river is 4505 km long, with a drainage area of 1 million km^2. The upper reaches of the Yangtze River have considerable falls, deep gorges, numerous tributaries and abundant precipitation. The average annual precipitation is about 1100 mm. The average

annual water volume at the outlet of Yichang station is 451 billion m^3, accounting for about 50% of the water volume of the Yangtze River flowing into the sea.

In this study, daily runoff data were collected from Yichang Station, Zhutuo Station and Pingshan Station, located in the upper reaches of the Yangtze River, as experimental data. The locations of the study watershed and stations are shown in Figure 4. The statistical information of these daily runoff data is shown in Table 1. The first 75% of all data were used as the training set, and the remaining 25% were used as the test set. Three daily runoff datasets are shown in Figure 5.

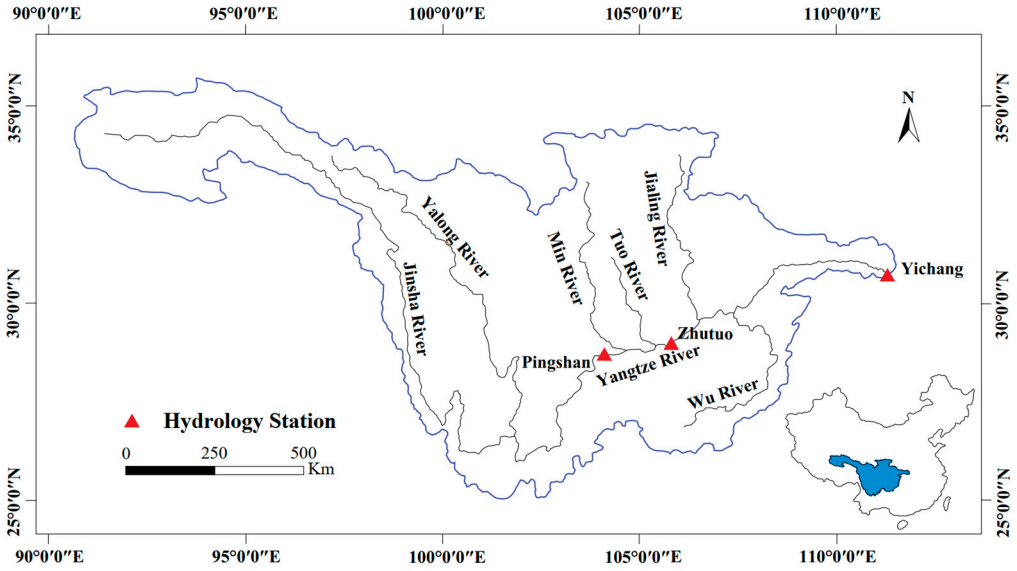

Figure 4. The locations of the studied watershed and three stations.

Table 1. Statistical information of three datasets.

Dataset	Station	Time	Mean	Maximum	Minimum	Standard Deviation
Dataset 1	Zhutuo	1 January 2000–31 December 2007	8283.83	37,400	2180	6325.06
Dataset 2	Yichang	1 January 1996–31 December 2003	13,781.37	61,700	2950	11,377.50
Dataset 3	Pingshan	1 January 2003–31 December 2010	4516.427	20,800	1180	3612.90

In this study, the MIC introduced above was used to quantify the correlation between historical runoff and current runoff, and historical runoff with an MIC value greater than 0.85 was selected as the input feature of the model. The calculation results of the MIC are shown in Table 2. The historical runoff selected as the model input feature is highlighted by gray filling. The number of model inputs for all models in dataset 1, dataset 2 and dataset 3 is 7, 5 and 6, respectively. The reason why this article uses three different time periods from three hydrological stations is that the length of the historical runoff series used as input for all models on the three datasets is inconsistent. Then, we can compare and analyze the impact of different quantities of model inputs on the training time of the same model.

Table 2. MIC values between historical runoff and current runoff.

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}	y_{t-10}
Dataset 1	0.969	0.940	0.909	0.887	0.874	0.861	0.852	0.841	0.827	0.815
Dataset 2	0.977	0.937	0.909	0.881	0.866	0.842	0.818	0.799	0.777	0.752
Dataset 3	0.960	0.931	0.905	0.884	0.870	0.855	0.841	0.829	0.822	0.819

Note: The historical runoff selected as the model input feature is highlighted by gray filling.

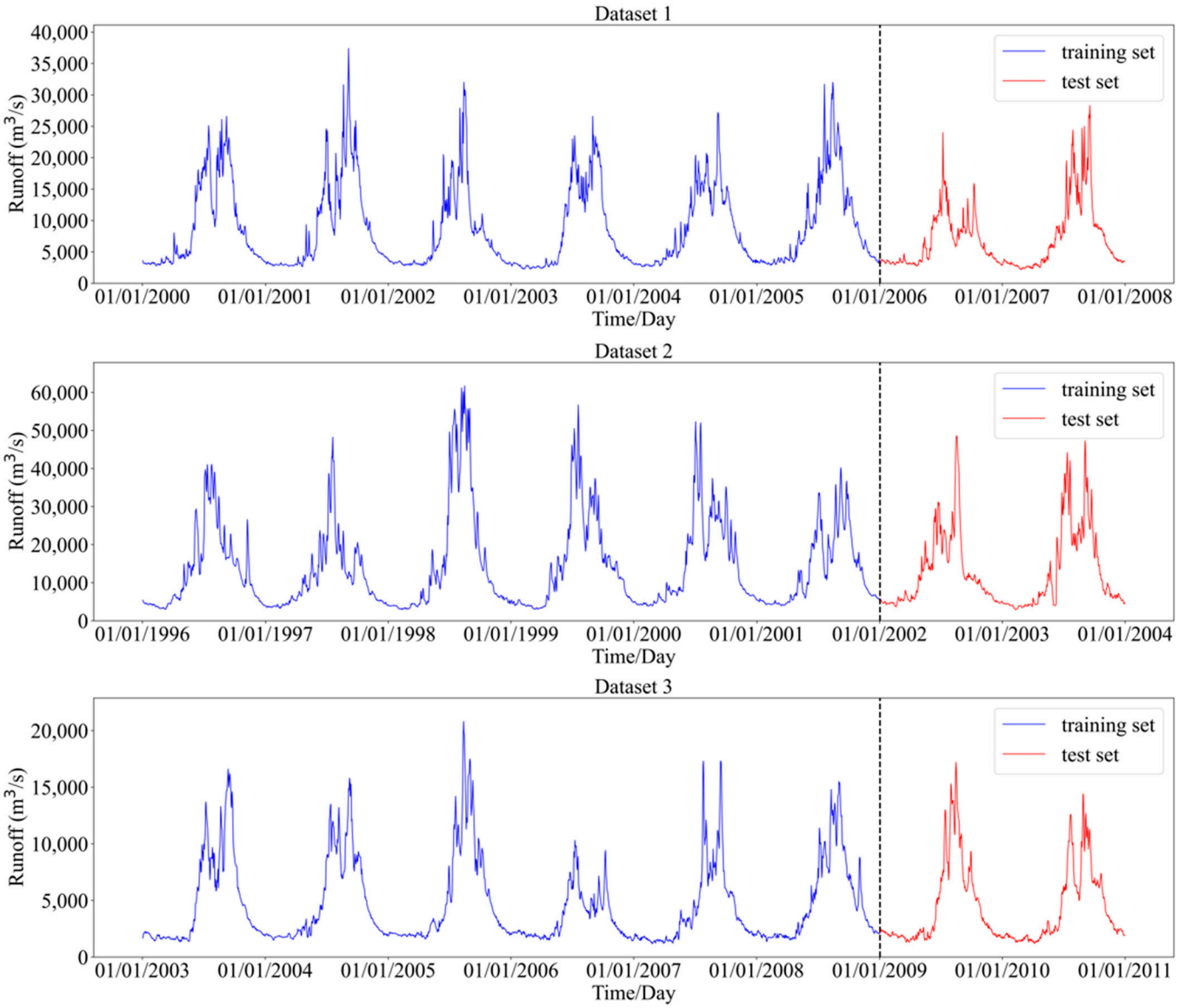

Figure 5. Three daily runoff datasets.

4.2. Experimental Design and Parameter Settings

The SMGM-NCQR model proposed in this article is a combined model based on the RNN model and probabilistic model. This type of combined model has been proven to be better than a single probabilistic model, for example, the combination models of GPR and RNN models [21], the combination models of DeepAR and RNN models [26] and the combination models of QR and RNN models [37]. Both RNN models and probabilistic models have a substantial impact on the performance of the combined model. This case study involved the completion of the following three tasks:

Task I: The different probabilistic models GPR, DeepAR, QR and NCQR proposed in this article were combined with the most common RNN model GRU to construct GRU-GPR, GRU-DeepAR, GRU-QR and GRU-NCQR, respectively, and comparisons were made in terms of four aspects, point prediction performance, interval prediction performance, probability density prediction performance and the degree of quantile crossing, to verify that the performance of the combination model based on NCQR is better than the combination model based on other probability models.

Task II: The different recurrent neural networks, GRU, MGM and SMGM, proposed in this article and NCQR were utilized to construct GRU-NCQR, MGM-NCQR and SMGM-

NCQR, respectively, which were compared in terms of four aspects: point prediction performance, interval prediction performance, probability density prediction performance and training time. The impact of different RNN models on NCQR-based models was compared.

Task III: The probability density curve predicted by SMGM-NCQR is displayed. For QR-based models and NCQR-based models, the outputs of these models are 19 quantiles with an interval of 0.05 between 0 and 1. KDE converts these predicted quantiles into continuous PDFs. GPR-based models and DeepAR-based models assume that daily runoff follows a Gaussian distribution and directly predict the parameters of the Gaussian distribution, namely mean and variance. By incorporating the predicted mean and variance into the Gaussian density function, the predicted PDFs can be obtained.

All model parameters were set using grid search results or commonly used values. To ensure a fair comparison, some of the same parameters in these models were equal. The batch size of all models was set to 64. In addition, all models were trained using the Adam optimizer with 100 epochs and a learning rate of 0.002. The parameter details of the six models in this article are shown in Table 3. Due to the randomness of the neural network model parameter initialization and training process, each model was run 10 times, and the average value was taken as the final result. In this study, all models were implemented based on the "pytorch 1.12.0" framework and the "scikit learn 1.1.1" package and run on a computer with an Intel (R) Core (TM) 3.60 GHz i7-9700 K CPU (Santa Clara, CA, USA) and NVIDIA GeForce GTX 1660 Ti (Santa Clara, CA, USA).

Table 3. The parameter details of the six models.

Model	Parameter	Value
GRU-GPR	number of GRU layer nodes	32
	number of GRU layers	4
	GPR kernel function	Rational quadratic kernel
	number of output layer nodes	2
GRU-DeepAR	number of GRU layer nodes	32
	number of GRU layers	4
	number of output layer nodes	2
GRU-QR	number of GRU layer nodes	32
	number of GRU layers	4
	number of output layer nodes	19
GRU-NCQR	number of GRU layer nodes	32
	number of GRU layers	4
	number of output layer nodes	20
MGM-NCQR	number of MGM layer nodes	32
	number of MGM layers	4
	number of output layer nodes	20
SMGM-NCQR	number of SMGM layer nodes	32
	number of SMGM layers	4
	number of output layer nodes	20
KDE	K-fold cross-validation in grid search for KDE bandwidth	5
	bandwidth range for KDE in grid search	(400, 450, 1)

4.3. Experimental Results and Comparative Analysis

4.3.1. Task I: Evaluation of Probability Models in the Combined Model

(1) It can be seen from Table 4 that GRU-DeepAR has the worst point prediction performance, with its RMSE and MAPE being 1045.69 m^3/s and 7.29%, respectively. The RMSE and MAPE of GRU-GPR are 1023.14 m^3/s and 6.35%, which are 9.67% and 12.9% higher than those of GRU-DeepAR, respectively. The RMSE and MAPE of GRU-QR are 955.81 m^3/s and 6.06%, respectively, and the RMSE and MAPE of GRU-NCQR are 949.98 m^3/s and 5.86%, respectively. The experimental results indicate that GRU-NCQR

has the smallest RMSR and MAPE, which means that GRU-NCQR has the best point prediction accuracy. In dataset 2 and dataset 3, the same conclusion can be obtained.

Table 4. Point prediction evaluation metrics of four models in Task I.

Models	Dataset 1		Dataset 2		Dataset 3	
	RMSE (m^3/s)	MAPE (%)	RMSE (m^3/s)	MAPE (%)	RMSE (m^3/s)	MAPE (%)
GRU-GPR	1023.14	6.35	1460.83	5.68	381.39	5.41
GRU-DeepAR	1132.67	7.29	1536.66	6.88	412.01	5.93
GRU-QR	955.81	6.06	1411.95	5.20	357.34	5.24
GRU-NCQR	949.98	5.86	1363.40	4.96	352.01	5.07

(2) It can be seen from Table 5 that the PICP of all interval predictions for all models on three datasets is greater than the corresponding confidence level. Taking dataset 1 as an example, the $PICP_{90\%}$ values of GRU-DeepAR, GRU-GPR, GRU-QR and GRU-NCQR are 0.9406, 0.9461, 0.9309 and 0.9448, respectively. The $PICP_{80\%}$ values of GRU-DeepAR, GRU-GPR, GRU-QR and GRU-NCQR are 0.9199, 0.8894, 0.8287 and 0.8240, respectively. The $PICP_{70\%}$ values of GRU-DeepAR, GRU-GPR, GRU-QR and GRU-NCQR are 0.8978, 0.7776, 0.7693 and 0.7127, respectively.

Table 5. Interval prediction evaluation metrics of the four models in Task I.

Models	Metrics	Dataset 1			Dataset 2			Dataset 3		
		90%	80%	70%	90%	80%	70%	90%	80%	70%
GRU-GPR	PICP	0.9406	0.9199	0.8978	0.9504	0.0872	0.8829	0.9655	0.9241	0.8966
	PINAW	0.1387	0.1081	0.0874	0.1119	0.0872	0.0705	0.1074	0.0780	0.0631
	CWC	0.1387	0.1081	0.0874	0.1119	0.0872	0.0705	0.1074	0.0780	0.0631
GRU-DeepAR	PICP	0.9461	0.8894	0.7776	0.9380	0.8815	0.8182	0.9448	0.8993	0.7917
	PINAW	0.1000	0.0670	0.0631	0.0842	0.0656	0.0530	0.0749	0.0583	0.0472
	CWC	0.1000	0.0670	0.0631	0.0842	0.0656	0.0530	0.0749	0.0583	0.0472
GRU-QR	PICP	0.9409	0.8523	0.7693	0.9159	0.8333	0.7479	0.9434	0.8428	0.7697
	PINAW	0.0826	0.0497	0.0375	0.0658	0.0432	0.0350	0.0723	0.0480	0.0374
	CWC	0.0826	0.0497	0.0375	0.0658	0.0432	0.0350	0.0723	0.0480	0.0374
GRU-NCQR	PICP	0.9448	0.8204	0.7127	0.9187	0.8292	0.7507	0.9641	0.8510	0.7379
	PINAW	0.0801	0.0379	0.0369	0.0677	0.0417	0.0322	0.0714	0.0457	0.0355
	CWC	0.0801	0.0379	0.0369	0.0677	0.0417	0.0322	0.0714	0.0457	0.0355

In dataset 1, the $PINAW_{90\%}$, $PINAW_{80\%}$ and $PINAW_{70\%}$ of GRU-GPR are 0.1387, 0.1081 and 0.0874, respectively, which are the maximum values among all compared models, indicating that GRU-GPR has the largest prediction interval. The $PINAW_{90\%}$, $PINAW_{80\%}$ and $PINAW_{70\%}$ of GRU-GPR are 0.1000, 0.0670 and 0.0631, respectively. Compared with GRU-DeepAR and GRU-GPR, GRU-QR reduces $PINAW_{90\%}$ by 39.64% and 13.27%, $PINAW_{80\%}$ by 53.6% and 25.1% and $PINAW_{70\%}$ by 56.67% and 39.94%, respectively. The results indicate that when predicting an unknown probability distribution, GPR and DeepAR as parametric methods provide a more conservative prediction interval compared to QR as a non-parametric method. That is, at the same confidence level, the prediction intervals of GRU-DeepAR and GRU-GPR are wider than that of GRU-QR. The $PINAW_{90\%}$, $PINAW8_{0\%}$ and $PINAW_{70\%}$ values of GRU-NCQR are 0.0801, 0.0379 and 0.0369, respectively, and the width of the interval prediction is smaller than those of the other comparison models in this task. In dataset 2 and dataset 3, GRU-NCQR also achieved the narrowest predicted interval and was compared with other models of Task I.

According to the definition of CWC, when the interval prediction PICP is greater than the corresponding confidence level, the penalty term is 0, so the CWC is equal to PINAW.

In short, the CWC of GRU-NCQR is smaller than those of the other comparison models in the three datasets, and GRU-NCQR can provide suitable interval prediction.

(3) From Table 6, it can be seen that the CS of GRU-GPR and GRU-DeepAR in all datasets is 0. This indicates that neither the GRU-GPR model nor the GRU-DeepAR model has quantile crossing. This is because GRU-GPR and GRU-DeepAR directly output the two parameters of the Gaussian distribution, namely mean and variance, and then directly obtain the Gaussian density function. The quantiles are obtained from the Gaussian density function, so there is no quantile crossing. GRU-QR can achieve better point prediction performance and interval prediction performance compared to GRU-GPR and GPR-DeepAR, but the CS of GRU-QR is not 0, indicating that GRU-QR has quantile crossing in the three datasets, which violates the monotonicity of conditional quantiles and seriously damages the reliability of prediction results. However, the CS of GRU-NCQR is 0 in all three datasets, indicating that NCQR ensures that the distance between any adjacent quantiles is greater than 0 by reconstructing the output of the QR to avoid quantile crossing.

Table 6. Quantifying indicators of quantile crossing degree for the four models in Task I.

Model	CS		
	Dataset 1	Dataset 2	Dataset 3
GRU-GPR	0	0	0
GRU-DeepAR	0	0	0
GRU-QR	267.71	425.36	124.15
GRU-NCQR	0	0	0

(4) According to Table 7, from dataset 1 to dataset 3, the CRPS values of GPU-GPR were 382.62, 539.07 and 177.90, respectively, which were the maximum values in each dataset, indicating that the probability density performance of GRU-GPR was the worst. The CRPS values of GRU-QR in the three datasets were 275.79, 401.60 and 141.28, respectively. Compared with GRU-DeepAR, they decreased by 10.25%, 8.79% and 7.65%, respectively. Compared with GRU-GRR, they decreased by 27.92%, 25.61% and 20.58%, respectively. The experimental results show that the probability density prediction performance of GRU-QR was higher than those of GRU-GPR and GRU-DeepAR because models based on GRP or DeepAR can only generate a Gaussian distribution function or Gaussian density function, and there may be significant differences between the actual distribution of runoff and Gaussian distributions. Meanwhile, QR and KDE are non-parametric models that can generate any form of probability distribution. The CRPS values of GRU-NCQR in the three datasets were 270.49, 389.40 and 138.42, respectively, which represent reductions of 5.30, 12.20 and 2.86 compared to GRU-QR, indicating that GRU-NCQR effectively avoids quantile crossing while improving the accuracy of probability density prediction.

To sum up, the various evaluation metrics of the combination model based on NCQR are better than those based on GPR, DeepAR and QR. This also verifies that in the combined model, NCQR achieves more reliable and excellent prediction performance compared to other probabilistic prediction models.

Table 7. Probability density prediction evaluation metric CRPS of the four models in Task I.

Models	CRPS		
	Dataset 1	Dataset 2	Dataset 3
GRU-GPR	382.62	539.07	177.90
GRU-DeepAR	307.30	440.29	152.99
GRU-QR	275.79	401.60	141.28
GRU-NCQR	270.49	389.40	138.42

4.3.2. Task II: Evaluation of RNN Models with NCQR-Based Models

The experimental results of Task I indicate that the model based on NCQR has a higher reliability and accuracy compared to the model based on other common probability models. In addition to probabilistic methods, RNN models that extract model input features also have a significant impact on the performance of NCQR-based models. This task involved comparing GRU-NCQR, MGM-NCQR and SMGM-NCQR to evaluate the impact of different RNN models on NCQR-based models.

(1) As shown in Table 8, in dataset 1, the RMSE of SMGM-NCQR is 916.78 m^3/s, representing reductions of 1.73% and 3.50%, respectively, compared with MGM-NCQR and GRU-NCQR. The MAPE of SMGM-NCQR is 4.87%, representing reductions of 8.11% and 16.89%, respectively, compared with MGM-NCQR and GRU-NCQR. This shows that the point prediction accuracy of SMGM-NCQR is higher than those of MGM-NCQR and GRU-NCQR because SMGM-NCQR achieves the smallest RMSE and smallest MAPE in Task II. The same conclusion can be obtained in dataset 2 and dataset 3. This indicates that compared to MGM and GRU, SMGM not only has a simpler model structure but can also better extract effective information from model inputs. Figure 6 shows the point prediction results of all models in Task II for the test sets of three datasets. In order to compare the point prediction results of different models more clearly, Figure 7 shows the point prediction results of all models in Task II for the first flood season of the three test sets, which is from June 10th to September 10th in the first year of each test set. From Figures 6 and 7, it can be observed that the severity of the lag phenomenon in SMGM-NCQR is lower than those of other comparative models, and the predicted runoff of SMGM-NCQR is closer to the actual observed runoff. These indicate that SMGM-NCQR has better performance in runoff point prediction.

Table 8. Point prediction evaluation metrics of four models in Task II.

Models	Dataset 1		Dataset 2		Dataset 3	
	RMSE (m^3/s)	**MAPE (%)**	**RMSE (m^3/s)**	**MAPE (%)**	**RMSE (m^3/s)**	**MAPE (%)**
GRU-NCQR	949.98	5.86	1363.40	4.96	352.01	5.07
MGM-NCQR	933.01	5.30	1347.63	4.77	346.78	4.66
SMGM-NCQR	916.78	4.87	1324.00	4.54	338.20	4.54

(2) As shown in Table 9, the PICP of all interval predictions for all models in the three datasets is greater than the corresponding confidence level, indicating that the results of interval prediction are valid.

In dataset 1, taking the interval prediction with the 90% confidence level as an example, the $PINAW_{90\%}$ of SMGM-NCQR is 0.0660, which is a reduction of 11.76% compared with MGM-NCQR and a reduction of 17.60% compared with GRU-NCQR. This shows that the predicted interval of SMGM-NCQR is narrower than those of MGM-NCQR and GRU-NCQR. In dataset 2, the $PINAW_{90\%}$, $PINAW_{80\%}$ and $PINAW_{70\%}$ of SMGM-NCQR are 0.0601, 0.0386 and 0.0601, respectively; the $PINAW_{90\%}$, $PINAW_{80\%}$ and $PINAW_{70\%}$ of MGM-NCQR are 0.0638, 0.0400 and 0.0298, respectively; and the $PINAW_{90\%}$, $PINAW_{80\%}$ and $PINAW_{70\%}$ of GRU-NCQR are 0.0677, 0.0417 and 0.0322, respectively, which shows that SMGM-NCQR has achieved the smallest PINAW in interval prediction at the three confidence levels. In dataset 3, similar conclusions can be obtained. The SMGM-NCQR achieves the narrowest interval width in the interval prediction in all three datasets while meeting the corresponding confidence level. This indicates that, compared with MGM-NCQR and GRU-NCQR, the prediction interval of SMGM-NCQR can provide more information to decision-makers at the same confidence level.

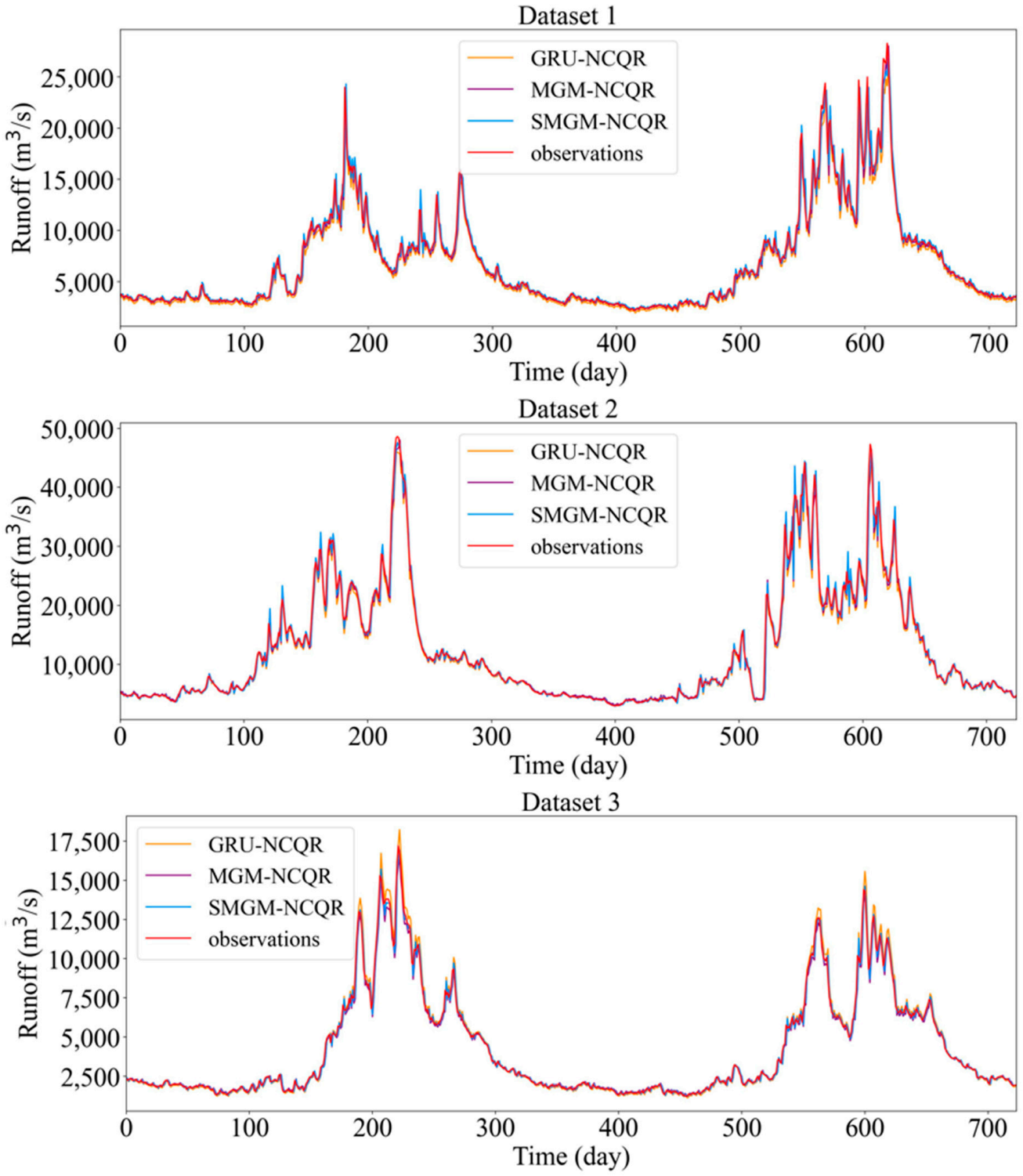

Figure 6. Point prediction results of four models in Task II on three test sets.

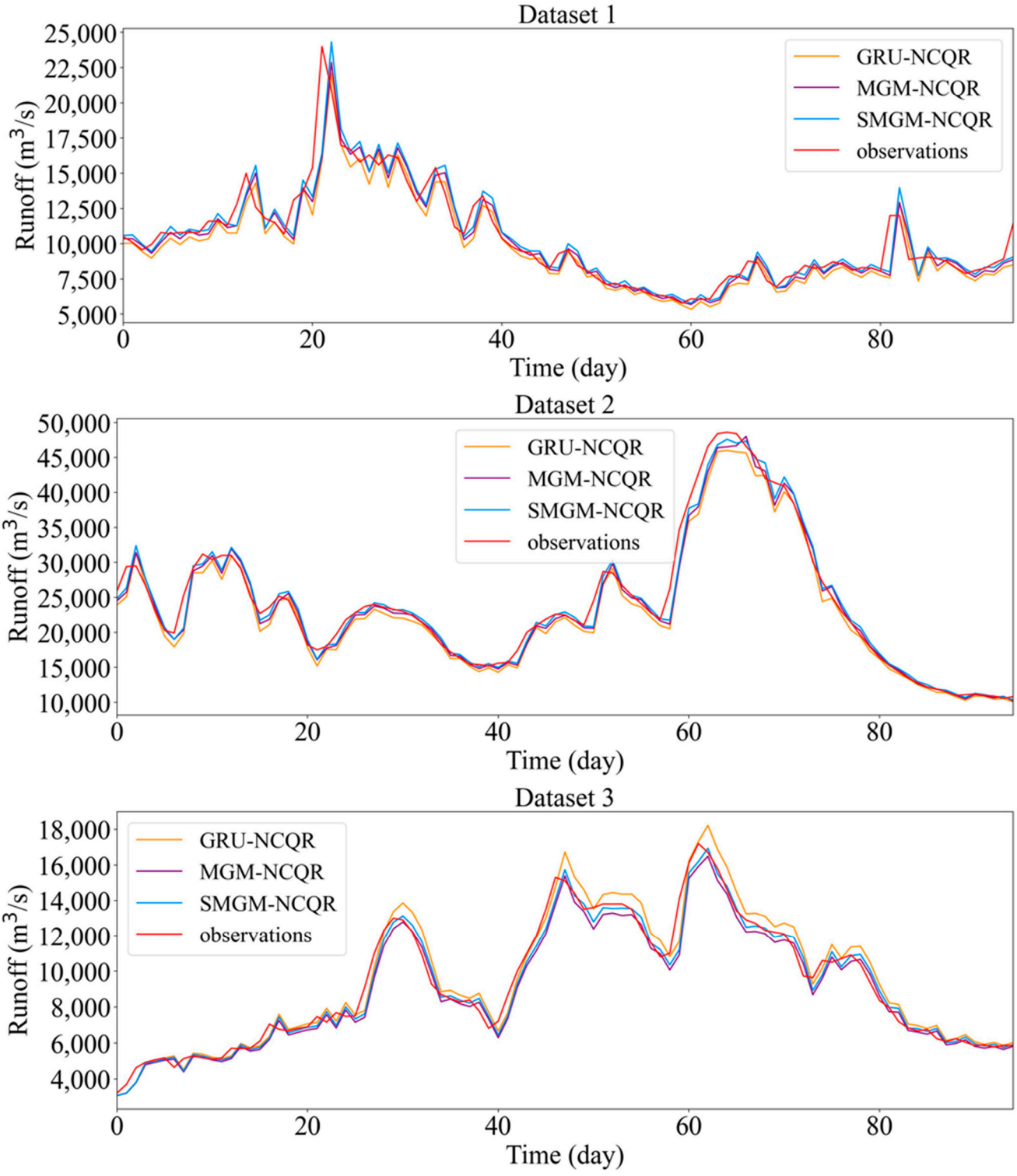

Figure 7. Point prediction results of four models in Task II during the flood season of three test sets.

The values of CWC and PINAW are numerically equal because the PICP of all interval predictions for all models in the three datasets is greater than the corresponding confidence level. The same conclusion as that for PINAW can be drawn, which is that the CWC of GRU-NCQR in Task II is the smallest. Figure 7 shows predicted intervals of SMGM-NCQR with confidence levels of 90%, 80% and 70% during the flood season of the first year in the test set of three datasets. From Figure 8, it can be seen that most observation points are within

the prediction interval with a 70% confidence level, and the interval width is relatively narrow, indicating that MGM-NCQR can provide an appropriate prediction interval.

Table 9. Interval prediction evaluation metrics of the three models in Task II.

Models	Metrics	Dataset 1			Dataset 2			Dataset 3		
		90%	80%	70%	90%	80%	70%	90%	80%	70%
GRU-NCQR	PICP	0.9448	0.8204	0.7127	0.9187	0.8292	0.7507	0.9641	0.8510	0.7379
	PINAW	0.0801	0.0379	0.0369	0.0677	0.0417	0.0322	0.0714	0.0457	0.0355
	CWC	0.0801	0.0379	0.0369	0.0677	0.0417	0.0322	0.0714	0.0457	0.0355
MGM-NCQR	PICP	0.9351	0.8412	0.7279	0.9215	0.8168	0.7410	0.9600	0.8524	0.7545
	PINAW	0.0748	0.0478	0.0359	0.0638	0.0400	0.0310	0.0671	0.0429	0.0326
	CWC	0.0748	0.0478	0.0359	0.0638	0.0400	0.0310	0.0671	0.0429	0.0326
SMGM-NCQR	PICP	0.9006	0.8204	0.7569	0.9229	0.8003	0.7066	0.9462	0.8234	0.7052
	PINAW	0.0660	0.0435	0.0327	0.0601	0.0386	0.0298	0.0654	0.0407	0.0312
	CWC	0.0660	0.0435	0.0327	0.0601	0.0386	0.0298	0.0654	0.0407	0.0312

(3) From Table 10, it can be seen that the CRPS of SMGM-NCQR in dataset 1 is 252.69, representing reductions of 3.11% and 6.59%, respectively, compared to MGM-NCQR and GRU-NCQR, which shows that SMGM-NCQR achieves the smallest CRPS in dataset 1. In dataset 2, the CRPS of SMGM-NCQR is 358.29, which is smaller than the CRPS of other comparison models. In dataset 3, the CRPS of SMGM-NCQR is 127.85, which is also the smallest CRPS. The results show that the probability density prediction performance of SMGM-NCQR is better than those of MGM-NCQR and GRU-NCQR.

Table 10. Probabilistic density evaluation metric of the three models in Task II.

Models	CRPS		
	Dataset 1	Dataset 2	Dataset 3
GRU-NCQR	270.49	389.40	138.42
MGM-NCQR	260.79	372.60	132.10
SMGM-NCQR	252.69	358.29	127.85

(4) As shown in Table 11, the training time of GRU-NCQR in the three datasets is 255 s, 198 s and 225 s, while the training time of MGM-NCQR in the three datasets is 198 s, 155 s and 174 s, respectively. Compared with the training time of GRU-NCQR, the training time of MGM-NCQR is reduced by 57 s, 43 s and 41 s, respectively. The reason is that the MGM model with only one gate has fewer parameters than the GRU model with two gates. The training time of SMGM-NCQR in the three datasets is 176 s, 139 s and 159 s, respectively, which are the minimum values in each dataset. This is attributed to SMGM simplifying the structure of MGM and further reducing the number of model parameters. As shown in Table 8, the number of model inputs for all models in dataset 2, dataset 3 and dataset 1 increases accordingly, with values of 5, 6 and 7, respectively. At the same time, the difference in training time between SMGM-NCQR and MGM-NCQR increases as the number of model inputs increases. This indicates that the difference in training time caused by a decrease in the number of model parameters will increase as the number of model inputs increases.

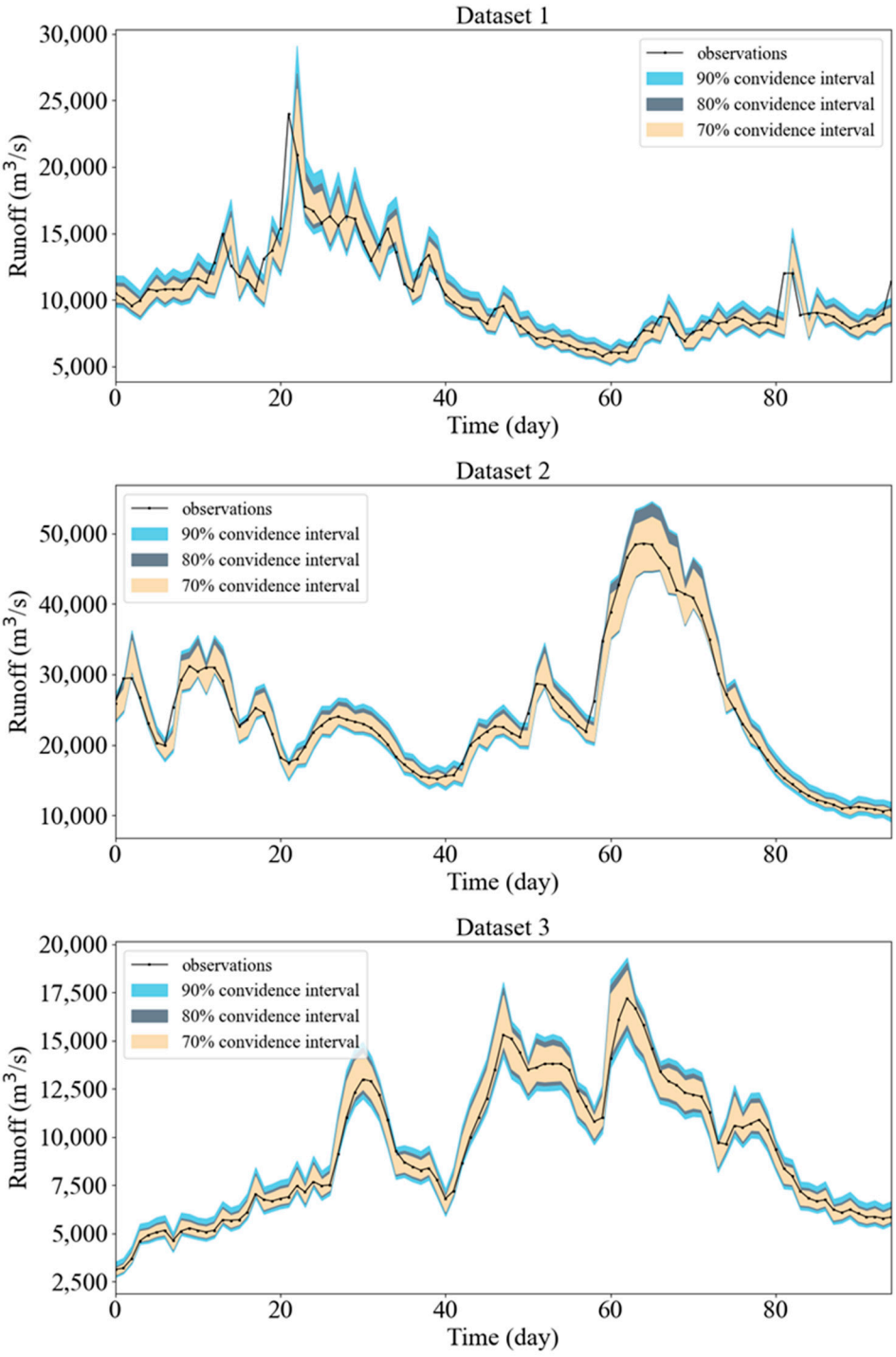

Figure 8. Predicted intervals of SMGM-NCQR with different confidence levels during the flood season of the three datasets.

Table 11. Training time of three models in Task II.

Models	Training Time (s)		
	Dataset 1	Dataset 2	Dataset 3
GRU-NCQR	255	198	225
MGM-NCQR	198	155	174
SMGM-NCQR	176	139	159

In summary, SMGM-NCQR spends the least training time and obtains the best prediction performance in all datasets.

4.3.3. Task III: Displaying the Probability Density Curve

The predicted probability density curves of SMGM-NCQR for nine periods selected at equal intervals in the test set of dataset 1 are shown in Figure 9. The period in Figure 9 represents the sequence number of the predicted runoff time series for the test set of dataset 1, with a day as the unit. The shapes of these nine probability density curves are not too wide or too narrow, which shows that these probability density curves are reasonable. The observed value of period 723 is almost at the peak of the probability density curve. In periods 1, 91, 362 and 452, the predicted value is near the peak of the probability density curve, which shows that the model achieves good prediction accuracy in these periods. The observed values in periods 181 and 272 are far away from the peak of the probability density curve, which indicates that the model achieves poor prediction accuracy in these periods. These probability density curves can comprehensively quantify the uncertainty of future daily runoff and provide more comprehensive and effective information for water resource planning and utilization.

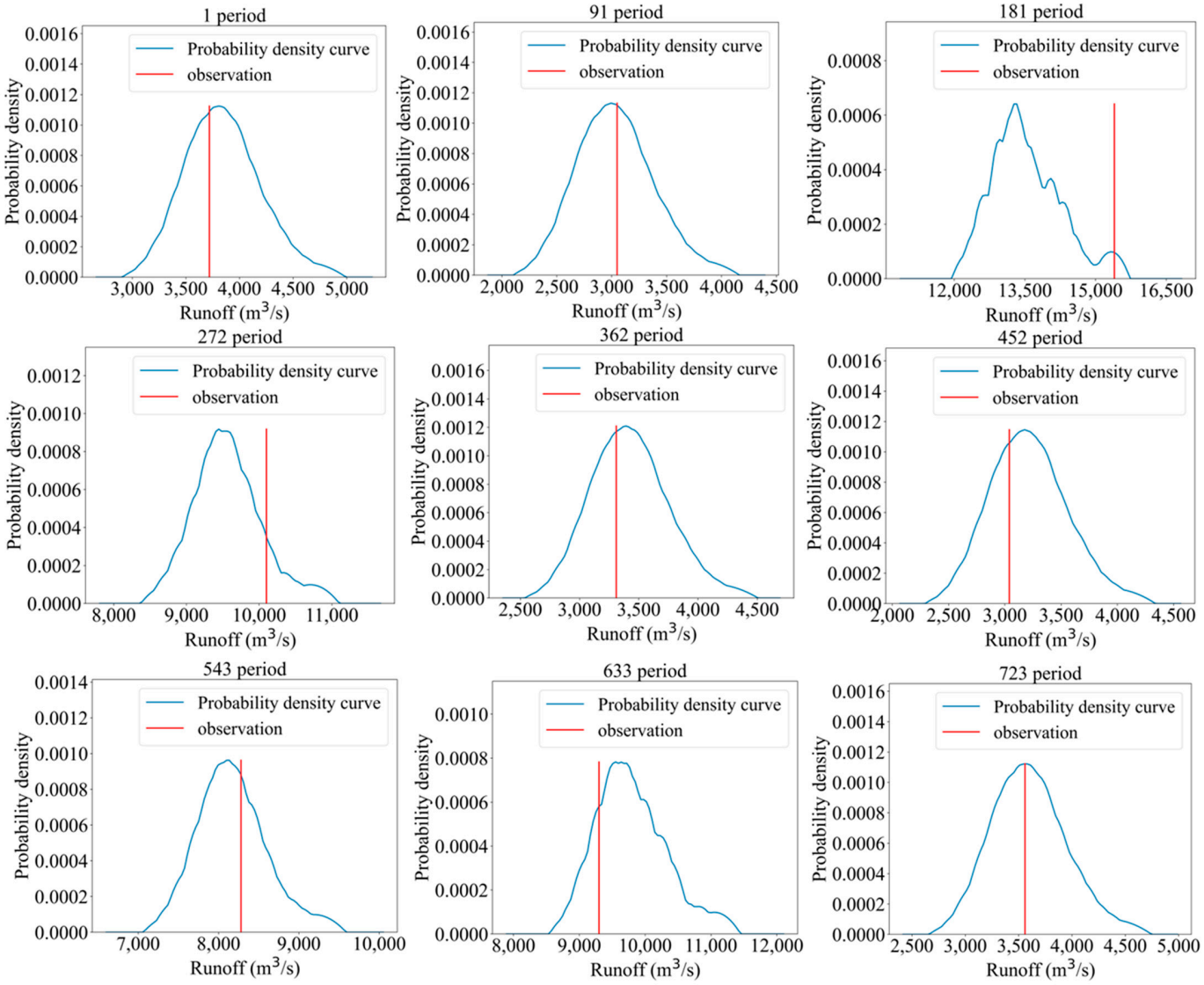

Figure 9. Nine probability density curves predicted by SMGM-NCQR in the test set of dataset 1.

5. Conclusions and Discussion

This article proposes a new probability density prediction model based on SMGM-NCQR and KDE. NCQR reconstructs the outputs of QR to ensure that the difference between any two adjacent quantiles is greater than 0, avoiding the occurrence of quantile crossing. In order to apply NCQR to prediction tasks of nonlinear daily runoff series, NCQR is combined with RNN models. In order to shorten the model training time and improve model accuracy, this paper simplifies the gate of MGM and proposes SMGM. Finally, this study uses KDE to convert the quantiles predicted by SMGM-NCQR into continuous PDFs. From the analysis of the experimental results, it can be concluded that the model proposed in this article has the following advantages:

(1) The NCQR proposed in this article avoids the common quantile crossing observed in QR-based models. At the same time, the prediction performance of the model based on NCQR is superior to that of models based on other probabilistic models.
(2) Among RNN models combined with NCQR, SMGM achieves the best predictive performance with the least training time compared to MGM and GRU. This indicates that SMGM not only has a simpler model structure but also has a better ability to extract effective information from model inputs compared with MGM and GRU.
(3) The new model based on SMGM-NCQR and KDE proposed in this article can efficiently obtain reliable and accurate probability density predictions of future daily runoff. While providing high-precision point predictions, it comprehensively quantifies the uncertainty of predictions, which can provide rich information for decision-makers in water conservancy systems.

However, our model has the following shortcomings:

(1) The input of the proposed SMGM-NCQR only includes historical runoff and does not consider other factors such as precipitation, daily maximum and minimum temperature. The probability density prediction model for daily runoff considering meteorological data is one of our future research directions.
(2) The parameters of SMGM-NCQR are calibrated based on the relationship between historical runoff and target runoff, ignoring the formation process of runoff. Although SMGM-NCQR has good probability density prediction performance, it is physically unknown and lacks interpretability. Improving the interpretability of SMGM-NCQR while maintaining high prediction accuracy is also one of our future research directions.

Author Contributions: Conceptualization, H.L.; methodology, H.L.; software, H.L.; validation, H.L., S.Z. and L.M.; investigation, H.L.; resources, H.L.; data curation, H.L.; writing—original draft preparation, H.L.; writing—review and editing, H.L., S.Z. and L.M.; visualization, H.L., S.Z. and L.M.; supervision, S.Z. and L.M.; project administration, S.Z. and L.M.; funding acquisition, S.Z. and L.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (No. 52379011 and No. 51979114).

Data Availability Statement: Data will be made available on request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, J.; Chen, X.; Khan, A.; Zhang, Y.-K.; Kuang, X.; Liang, X.; Taccari, M.L.; Nuttall, J. Daily runoff forecasting by deep recursive neural network. *J. Hydrol.* **2021**, *596*. [CrossRef]
2. Amiri, E. Forecasting daily river flows using nonlinear time series models. *J. Hydrol.* **2015**, *527*, 1054–1072. [CrossRef]
3. Zhang, J.; Yan, H. A long short-term components neural network model with data augmentation for daily runoff forecasting. *J. Hydrol.* **2023**, *617*, 128853. [CrossRef]
4. Wang, W.-C.; Chau, K.-W.; Cheng, C.-T.; Qiu, L. A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. *J. Hydrol.* **2009**, *374*, 294–306. [CrossRef]
5. Liu, G.; Tang, Z.; Qin, H.; Liu, S.; Shen, Q.; Qu, Y.; Zhou, J. Short-term runoff prediction using deep learning multi-dimensional ensemble method. *J. Hydrol.* **2022**, *609*, 127762. [CrossRef]

6. Yuan, X.; Chen, C.; Lei, X.; Yuan, Y.; Muhammad Adnan, R. Monthly runoff forecasting based on LSTM–ALO model. *Stoch. Environ. Res. Risk Assess.* **2018**, *32*, 2199–2212. [CrossRef]
7. Wu, J.; Wang, Z.; Hu, Y.; Tao, S.; Dong, J. Runoff forecasting using convolutional neural networks and optimized bi-directional long short-term memory. *Water Resour. Manag.* **2023**, *37*, 937–953. [CrossRef]
8. Lu, M.; Hou, Q.; Qin, S.; Zhou, L.; Hua, D.; Wang, X.; Cheng, L. A Stacking Ensemble Model of Various Machine Learning Models for Daily Runoff Forecasting. *Water* **2023**, *15*, 1265. [CrossRef]
9. Singh, R.; Subramanian, K.; Refsgaard, J.C. Hydrological modelling of a small watershed using MIKE SHE for irrigation planning. *Agric. Water Manag.* **1999**, *41*, 149–166. [CrossRef]
10. Baker, T.J.; Miller, S.N. Using the Soil and Water Assessment Tool (SWAT) to assess land use impact on water re-sources in an East African watershed. *J. Hydrol.* **2013**, *486*, 100–111. [CrossRef]
11. Zhang, Q.; Wang, B.D.; He, B.; Peng, Y.; Ren, M.L. Singular spectrum analysis and ARIMA hybrid model for annual runoff forecasting. *Water Resour. Manag.* **2011**, *25*, 2683–2703. [CrossRef]
12. Valipour, M. Long-term runoff study using SARIMA and ARIMA models in the United States. *Meteorol. Appl.* **2015**, *22*, 592–598. [CrossRef]
13. Gizaw, M.S.; Gan, T.Y. Regional flood frequency analysis using support vector regression under historical and future climate. *J. Hydrol.* **2016**, *538*, 387–398. [CrossRef]
14. Ehteram, M.; Afan, H.A.; Dianatikhah, M.; Ahmed, A.N.; Ming Fai, C.; Hossain, M.S.; Elshafie, A. Assessing the predictability of an improved ANFIS model for monthly streamflow using lagged climate indices as predic-tors. *Water* **2019**, *11*, 1130. [CrossRef]
15. Xu, Z.; Zhou, J.; Mo, L.; Jia, B.; Yang, Y.; Fang, W.; Qin, Z. A Novel Runoff Forecasting Model Based on the Decomposition-Integration-Prediction Framework. *Water* **2021**, *13*, 3390. [CrossRef]
16. Xu, Y.; Hu, C.; Wu, Q.; Jian, S.; Li, Z.; Chen, Y.; Wang, S. Research on particle swarm optimization in LSTM neural networks for rainfall-runoff simulation. *J. Hydrol.* **2022**, *608*, 127553. [CrossRef]
17. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
18. Rahimzad, M.; Moghaddam Nia, A.; Zolfonoon, H.; Soltani, J.; Danandeh Mehr, A.; Kwon, H.H. Performance comparison of an LSTM-based deep learning model versus conventional machine learning algorithms for streamflow fore-casting. *Water Resour. Manag.* **2021**, *35*, 4167–4187. [CrossRef]
19. Cho, K.; Van Merrienboer, B.; Bahdanau, D.; Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv* **2014**, arXiv:1409.1259.
20. Gao, S.; Huang, Y.; Zhang, S.; Han, J.; Wang, G.; Zhang, M.; Lin, Q. Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation. *J. Hydrol.* **2020**, *589*, 125188. [CrossRef]
21. Zhang, Z.; Qin, H.; Liu, Y.; Yao, L.; Yu, X.; Lu, J.; Jiang, Z.; Feng, Z. Wind speed forecasting based on quantile regression minimal gated memory network and kernel density estimation. *Energy Convers. Manag.* **2019**, *196*, 1395–1409. [CrossRef]
22. Zhang, Z.; Tang, H.; Qin, H.; Luo, B.; Zhou, C.; Zhou, H. Multi-step ahead probabilistic forecasting of multiple hydrological variables for multiple stations. *J. Hydrol.* **2023**, *617*, 129094. [CrossRef]
23. Faucher, D.; Rasmussen, P.F.; Bobée, B. A distribution function based bandwidth selection method for kernel quan-tile estimation. *J. Hydrol.* **2001**, *250*, 1–11. [CrossRef]
24. Sun, A.Y.; Wang, D.; Xu, X. Monthly streamflow forecasting using gaussian process regression. *J. Hydrol.* **2014**, *511*, 72–81. [CrossRef]
25. Bai, H.; Li, G.; Liu, C.; Li, B.; Zhang, Z.; Qin, H. Hydrological probabilistic forecasting based on deep learning and Bayesian optimization algorithm. *Hydrol. Res.* **2021**, *52*, 927–943. [CrossRef]
26. Zou, Y.; Wang, J.; Lei, P.; Li, Y. A novel multi-step ahead forecasting model for flood based on time residual LSTM. *J. Hydrol.* **2023**, *620*, 129521. [CrossRef]
27. Tareghian, R.; Rasmussen, P.F. Statistical downscaling of precipitation using quantile regression. *J. Hydrol.* **2013**, *487*, 122–135. [CrossRef]
28. Papacharalampous, G.; Langousis, A. Probabilistic water demand forecasting using quantile regression algorithms. *Water Resour. Res.* **2022**, *58*, e2021WR030216. [CrossRef]
29. Fan, Y.R.; Huang, G.H.; Li, Y.P.; Wang, X.Q.; Li, Z. Probabilistic prediction for monthly streamflow through coupling stepwise cluster analysis and quantile regression methods. *Water Resour. Res. Manag.* **2016**, *30*, 5313–5331. [CrossRef]
30. Regression, Q. *Handbook of Quantile Regression*; CRC Press: Boca Raton, FL, USA, 2017.
31. Jahangir, M.S.; You, J.; Quilty, J. A quantile-based encoder-decoder framework for multi-step ahead runoff fore-casting. *J. Hydrol.* **2023**, *619*, 129269. [CrossRef]
32. Wang, Y.; Gan, D.; Sun, M.; Zhang, N.; Lu, Z.; Kang, C. Probabilistic individual load forecasting using pinball loss guided LSTM. *Appl. Energy* **2019**, *235*, 10–20. [CrossRef]
33. Benson, D.A.; Bolster, D.; Pankavich, S.; Schmidt, M.J. Nonparametric, data-based kernel interpolation for particle-tracking simulations and kernel density estimation. *Adv. Water Resour.* **2021**, *152*, 103889. [CrossRef]
34. He, Y.; Li, H. Probability density forecasting of wind power using quantile regression neural network and kernel density estimation. *Energy Convers. Manag.* **2018**, *164*, 374–384. [CrossRef]
35. Epanechnikov, V.A. Non-parametric estimation of a multivariate probability density. *Theory Probab. Its Appl.* **1969**, *14*, 153–158. [CrossRef]

36. Reshef, D.N.; Reshef, Y.A.; Finucane, H.K.; Grossman, S.R.; McVean, G.; Turnbaugh, P.J.; Lander, E.S.; Mitzenmacher, M.; Sabeti, P.C. Detecting novel associations in large data sets. *Science* **2011**, *334*, 1518–1524. [CrossRef] [PubMed]
37. Sun, Q.; Tang, Z.; Gao, J.; Zhang, G. Short-term ship motion attitude prediction based on LSTM and GPR. *Appl. Ocean Res.* **2022**, *118*, 102927. [CrossRef]