Monthly Streamflow Forecasting Using Convolutional Neural Network

Xingsheng Shu
Dalian University of Technology
Wei Ding
Dalian University of Technology
Yong Peng (pyongcuidi@163.com)
Dalian University of Technology
Ziru Wang
Dalian University of Technology
Jian Wu
Dalian University of Technology
Min Li
Dalian University of Technology

Research Article

Keywords: monthly streamflow forecasting, atmospheric circulation factors, inputs selection, artificial intelligence, convolutional neural network

Posted Date: July 6th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-239777/v1

Version of Record: A version of this preprint was published at Water Resources Management on November 1st, 2021. See the published version at https://doi.org/10.1007/s11269-021-02961-w.

Monthly Streamflow Forecasting Using Convolutional Neural Network

Xingsheng Shu, Wei Ding*, Yong Peng*, Ziru Wang, Jian Wu, Min Li

Faculty of Infrastructure Engineering, Dalian University of Technology, Dalian, Liaoning 116023, China

*Corresponding author 1: Yong Peng, Email: pengyong@dlut.edu.cn

*Corresponding author 2: Wei Ding, Email: weiding@dlut.edu.cn

Highlights:

- The convolutional neural network is investigated for monthly streamflow forecasting.
- The input selection process can be automatically completed by the convolutional neural network.
- The performance of the CNN is superior to the ANN and ELM with smaller errors and better stability.

Abstract: Monthly streamflow forecasting is vital for the management of water resources. Recently, numerous studies have explored and evidenced the potential of artificial intelligence (AI) models in hydrological forecasting. In the current study, the feasibility of a relatively new AI model, namely the convolutional neural network (CNN), is explored for forecasting monthly streamflow. The CNN is a method of deep learning, the unique convolution-pooling mechanism in which creates its superior attribute of automatically extracting critical features from input layers. Hydrological and large-scale atmospheric circulation variables including rainfall, streamflow, and atmospheric circulation factors (ACFs) are used to establish models and forecast streamflow for Huanren Reservoir and Xiangjiaba Hydropower Station, China. The ANN and ELM with inputs identified based on cross-correlation analysis (CC) and mutual information analysis (MI) are established for comparative analysis. The performances of these models are assessed with several statistical metrics and graphical evaluation methods. The results show that CNN performs better than ANN and ELM across all the statistical measures. Moreover, CNN shows better stability in forecasting accuracy.

Keywords: monthly streamflow forecasting, atmospheric circulation factors, inputs selection, artificial intelligence,

convolutional neural network

1 Introduction

Long-term hydrological streamflow forecasting has been considered an important basis for the design, construction, and management of water conservancy and hydropower projects. This type of forecasting has a long forecast period, which gives water managers sufficient time to allocate water to different sectors. But the complex nature of the processes such as mechanism of runoff generation, climatic variation, the effect of human activity make it difficult to simulate and forecast them with desirable accuracy. Therefore, favorable long-term streamflow forecasting is a challenging task (Yaseen et al. 2016).

During the past few decades, numerous methods have been developed to forecast long-term streamflow (Zhang et al. 2015). These models can be classified into physical models and data-driven models (Sahay and Srivastava 2014). Typically, physical models have the advantage of assisting the physical understanding of the hydrological process, but complex physical equations, parametric assumptions, and the variables involved in the modeling process make them difficult to be designed and implemented (Wu and Chau 2006; Yaseen et al. 2016). Data-driven models find relationships between system state variables without explicit knowledge of the physical behavior using statistic or machine learning algorithms (Ghorbani et al. 2016). Due to their advantages including simplicity in design and implementation, minimum information requirements, and relatively high accuracy, the data-driven models are becoming increasingly popular in hydrological forecasting (Yaseen et al. 2016; Adamowski and Sun 2010).

Classical regression techniques such as autoregressive model (AR), moving average model (MA) and autoregressive moving average (ARMA) model may be the earlier tools that have been used in streamflow forecasting (Abrahart and See 2000; Montanari et al. 2000; YU and TSENG 1996). But the assumptions of stationary and linearity have limited their capacity in capturing the non-linear pattern of streamflow (Yaseen et al. 2019). To overcome this drawback, AI models with

47 the ability in nonlinear mapping have been vastly developed as alternatives in the last several decades.

48 The ANN is one of the most popular examples of AI techniques since the 1990s, and it has been applied in various
49 areas of water-related research, particularly in streamflow forecasting (Samsudin et al. 2011; Afan et al. 2020). However,
50 ANN also has some drawbacks. As an example, the determination of its optimal network structure is difficult and usually
51 requires a trial-and-error process (Ozgur Kisi 2004). The SVM proposed by (Vapnik 1995) is a more advanced AI model,
52 which is based on the principle of structural risk minimization, theoretically minimizes the expected error of a learning
53 machine and therefore reduces the phenomenon of overfitting (Yu et al. 2017). This AI model has gained the attention of
54 many researchers and been applied to streamflow forecasting with promising results (LIN et al. 2006; Samsudin et al. 2011;
55 Shabri and Suhartono 2012). However, this method has the disadvantages of time-consuming and facing a high
56 computational burden. As an emerging class of ANN, ELM proposed by (Huang et al. 2006) is another powerful tool. In
57 this model, the input weights and hidden biases are randomly initialized, and the output weights are directly calculated by
58 the least squares method. Therefore, it has a fast learning speed for its iterative-free learning mechanism. Moreover, this
59 model overcomes some disadvantages including over-fitting and stucking in local minima that some traditional models
60 suffer from (Yaseen et al. 2016). The ELM has been successfully used in streamflow forecasting and proven to have good
61 generalization capability (Li and Cheng 2014; Yaseen et al. 2016; Deo and Şahin 2016). But there is still room for
62 improvement. Like some other ANN-based models, the random assignment of weight parameters may cause inferior values
63 (Li et al. 2016).

64 Although a lot of research efforts have been devoted to improve the reliability and accuracy of streamflow forecasting,
65 up to date, there has been no approach that can achieve the best accuracy for all catchments (Fu et al. 2020). Sometimes,
66 one model outperforms other models in one catchment may not exhibit good accuracy in another. This phenomenon may
67 be caused by some complex physical processes that characterize different catchments and directly affect the streamflow
68 datasets (Yaseen et al. 2019; Fu et al. 2020). Given this situation, it is necessary for scholars to continuously investigate the

reliability in streamflow forecasting of newly developed models. Therefore, this study attempts to investigate the potential of another AI approach, CNN, for monthly streamflow forecasting. Compared to the ANN, CNN adds the convolution, pooling, and fully connected layers and thus has much deeper and more complex architectures. The add of such layers allows CNN to extract critical features from the input layer automatically without prior knowledge and human effort. According to the form of the input layer, CNN currently used in the field of hydrology can be designed into one-dimensional or two-dimensional CNN (namely 1D-CNN and 2D-CNN). The 1D-CNN involves vectors as inputs, while 2D-CNN employs two-dimensional matrixes as inputs. In this study, we use the 2D-CNN for one-step-ahead monthly streamflow forecasting due to its stronger feature extracting capability. This capability makes the 2D-CNN immune to the step of selecting appropriate input variables from a large amount of historical data time series, which is the merit of 2D-CNN.

During the past two years, CNN has been gradually used in hydrological forecasting (Huang et al. 2020; Haidar and Verma 2018; Chong et al. 2020; Hussain et al. 2020). But these researches mainly focused on the applications of 1D-CNN. (Haidar and Verma 2018) applied a 1D-CNN to forecast monthly rainfall for a suburb in eastern Australia. The results evidenced its good capacity in monthly rainfall forecasting. (Chong et al. 2020) developed a hybrid 1D-CNN coupled with a wavelet transform technique to forecast daily and monthly rainfall over the Langat River Basin in Malaysia. The results showed that the 1D-CNN could capture patterns of the rainfall time series for both monthly forecasting and daily forecasting. (Hussain et al. 2020) performed a study on the 1D-CNN to forecast the streamflow for four rivers in the UK. Compared to 1D-CNN, the 2D-CNN is rarely investigated. (Huang et al. 2020) applied a 2D-CNN to forecast daily streamflow and the forecasting accuracy of the proposed 2D-CNN was reported to be much better than that of comparative models.

Motivated by its automatically feature extracting ability, outperformance in daily streamflow forecasting, and limited applications in streamflow forecasting, this paper attempts to investigate the potential of the 2D-CNN to accurately forecast monthly streamflow. Antecedent rainfall, streamflow, and ACFs data are converted into two-dimensional matrixes and they are utilized to drive CNNs to forecast one-month-ahead streamflow, which, to the best of our knowledge, has not been

conducted by others to date. The ANN and ELM are chosen as comparative methods. Two cases, Huanren Reservoir and Xiangjiaba Hydropower Station of China are used to verify the feasibility of CNN.

2 Theoretical Overview

2.1 Convolutional neural network

CNN is an efficient image processing algorithm. Generally, CNN refers to a two-dimensional convolutional neural network, the inputs of which are two-dimensional matrixes (Hussain et al. 2020). However, there are also other types of CNN such as one-dimensional CNN and three-dimensional CNN. All types of CNN share the same characteristics but differ in the dimension of the input matrix and the way the filters slide over the data. The CNN employed in this study is the two-dimensional CNN. Two salient characteristics contribute to the uniqueness of CNN. First, a neuron is only connected to their local nearby neurons in the previous layer. Second, the pooling mechanism is introduced to significantly reduce the number of coefficients in the network while preserving the most important features scanned from its receptive region (Mozo et al. 2018). A standard CNN is generally composed of five types of layers: input layer, convolution layer, pooling layer, fully connected layer, and output layer, as shown in Fig. 1.

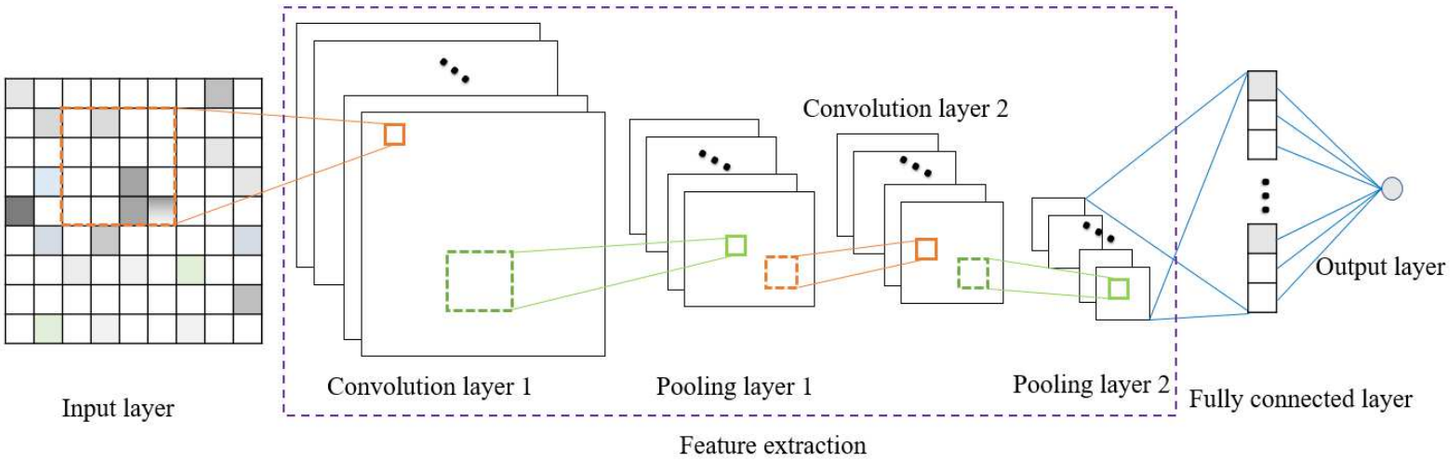

Fig. 1 Typical convolutional network architecture for image recognition

The input layer provides input information for the whole model and is usually a two-dimensional matrix. In this study, antecedent data with time and space dimensions are employed to forecast streamflow. Let *x*- and *y*-axis denote the time and space of a matrix, respectively. The elements in the matrix are observations of different input variables at different times. In the time dimension, time ranges from the past to the present, and the time interval, which is usually 1 hour, 1 day, or 1 month, depends on the type of forecasting. For the one-month-ahead streamflow forecasting, the time interval is 1 month. In the space dimension, different variables are viewed as a sequence of dots. Suppose the length and width of the input matrix are *m* and *n*, respectively, the input matrix *X* can be written as:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & x_{ij} & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{1}$$

where *i* denotes different time series, *j* denotes the periods, *m* represents the number of time series that are employed, *n* is the length of time series; x_{ij} represents the value of *i*th time series at the *j*th period.

The convolution layer is the core building block of a CNN (Haidar and Verma 2018). This layer is composed of several convolution kernels and aims to learn feature representations from the input layer. The outputs of a convolution layer are passed through a linear operation and a nonlinear activation function processing before being fed into the next layer. Denote the parameters, input, and output of the *j*th kernel in the *l*th convolution layer by (ω_l^j, b_l^j), x_l and o_l^j, respectively, the output of the *j*th kernel can be written as:

$$o_l^j = f\left(\sum_{k=1}^{M_l-1} \omega_l^j x_l^k + b_l\right), \quad j \in [1, M_l] \tag{2}$$

where *j* is the index of the convolution kernel in the convolution layer, x_l^k is the *i*th channel of the x_l, M_l is the number of kernels in the *l*th layer; $f(\cdot)$ is the activation function and b_l^j represents a bias item.

The pooling layer is used to decrease the size of the outputs from convolution layers with pooling methods, such as max-pooling and average-pooling. The max-pooling layer selects the highest value in each scanned region of its inputs

whereas the average-pooling layer calculates the average value. In this study, the average-pooling operation is employed. Since only one value, the maximum or the average value, in each scanned region are selected, the number of CNN parameters after the pooling operation is greatly reduced. The alternation of the convolution and pooling layers can not only reduce the network scale of a CNN but also identify the most prominent features of input layers.

In the fully connected layer, different features learned by the convolution and pooling layers are converted into a dense vector. The output layer is used to establish the equation $y = Wo + b$ among the dense vector o, bias item b, and the forecasted value y. Once these parameters (the vector W and b) are determined, a final forecasted result can be obtained when giving a dense vector o.

2.2 Artificial neural network

ANN is a nonlinear dynamic system whose original development was based on simulating the structure and function of the human brain (Imrie et al. 2000). This method was reported that can approximate any nonlinear mathematical input-output relations (Cichocki and Unbehauen 1993; Pashova and Popova 2011). There are many variants of ANN (Yılmaz and Yuksek 2008). In this study, we employ the feed-forward backpropagation (BP) network as a comparative model. The training of a BP neural network is an optimization process (Kuang and Xu 2018), which consists of two parts: the forward pass and the backward pass. In the forward pass, the input is processed through the ANN, and then the forecasted results are obtained. If the deviation between the forecast and the observation is large, the backward pass will be carried out to modify the parameters of each layer. After enough iterations of the forward and backward pass, an ANN model will be able to make forecasting correctly.

2.3 Extreme learning machine

The ELM was first proposed by (Huang et al. 2006) as a learning algorithm for a single hidden layer feed-forward neural networks (Hadi et al. 2020). This model exhibits several important advantages, which make it an appealing method

for streamflow forecasting (Li et al. 2016). Here, we present a brief description of the ELM.

Considering a set of training samples $\{(X_1,y_1),(X_2,y_2),\cdots,(X_i,y_i),\cdots,(X_N,y_N)\}$, where $X_i \subset R^l$ is the input vector of the *i*th sample, y_i is the corresponding observation, and *N* is the size of training samples. The ELM with *P* hidden neurons can be modeled as:

$$\sum_{j=1}^{P} \beta_j f(\omega_j X_i + b_j) = y_i, \quad i = 1, 2, \cdots, N \tag{3}$$

where ω_j is the weight vector connecting the input variables to the *j*th hidden neuron, β_j is the weight vector connecting the *j*th hidden neuron to the output neuron, and b_j is the bias of the *j*th hidden neuron. The Eq.(3) can be expressed in matrix form as $\boldsymbol{y} = \mathbf{H}\boldsymbol{\beta}$, where $\boldsymbol{y} = (y_1,y_2,\cdots,y_N)^T$, $\boldsymbol{\beta} = (\beta_1,\beta_2,\cdots,\beta_P)^T$ and $\mathbf{H}$ defined by:

$$\mathbf{H} = \begin{bmatrix} f(\omega_1 X_1 + b_1) & \cdots & f(\omega_P X_1 + b_P) \\ \vdots & \ddots & \vdots \\ f(\omega_1 X_N + b_1) & \cdots & f(\omega_P X_N + b_P) \end{bmatrix} \tag{4}$$

The solution of the ELM is $\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\boldsymbol{y}$, where $\mathbf{H}^{\dagger} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}$ is called Moore-Penrose generalized inverse of $\mathbf{H}$ (Huang et al. 2004). Finally, the forecasted value is given by:

$$\hat{y} = \sum_{j=1}^{P} \hat{\boldsymbol{\beta}} f(\omega_j X_{test} + b_j) \tag{5}$$

where X_{test} is the input vector in the testing period. For a more detailed evaluation of the ELM algorithm, readers can refer to the studies of (Huang et al. 2006).

2.4 Input selection methods for ANN and ELM

In this study, two widely used input selection techniques, cross-correlation analysis and mutual information analysis, are used to determine the most informative variables. The cross-correlation function measures how two random variables *X* and *Y* co-vary linearly by calculating the linear correlation coefficient r_{XY}:

$$r_{XY} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{6}$$

where x_i, y_i are sample values of X and Y, N is the sample size, and $\bar{x}$, $\bar{y}$ are the mean value of the samples. $r_{XY} = 1$ means a perfect linear correlation while an intermediate value corresponds to partial correlations, and $r_{XY} = 0$ means X is uncorrelated with Y.

MI is a measure that quantifies the stochastic dependency between two random variables without making any assumptions about the nature of their relation (Babel et al. 2015). It can capture more general relationships between variables (Quilty et al. 2016). In the case of two discrete variables X and Y, the index MI can be expressed as follows (Siqueira et al. 2018):

$$MI(X,Y) = \sum\sum p(X,Y)\log\left(\frac{p(X,Y)}{p(X)\,p(Y)}\right) \tag{7}$$

where the $p(X,Y)$ is the joint probability density function, $p(X)$ and $p(Y)$ are the marginal distribution function of X and Y, respectively. The value of MI ranges between 0 and infinity (∞). $MI(X,Y) = 0$ means that X and Y are independent of each other. If the X and Y are dependent, the MI value will be greater than 0, and the larger the value, the stronger the dependence.

2.5 Model performance evaluation

The following three statistical indices are used to evaluate the performance of the models developed in this study (Hadi et al. 2019).

I. Mean absolute error (*MAE*), expressed as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Q_{sim}^{i} - Q_{obs}^{i}| \tag{8}$$

II. Root mean square error (*RMSE*), expressed as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Q_{sim}^{i} - Q_{obs}^{i})^{2}} \tag{9}$$

III. Nash-Sutcliffe efficiency coefficient (*NSE*), expressed as:

$$NSE = 1 - \frac{\sum_{i=1}^{N}(Q_{obs}^{i} - Q_{sim}^{i})^{2}}{\sum_{i=1}^{N}\left(Q_{obs}^{i} - \bar{Q}_{obs}\right)^{2}} \tag{10}$$

where Q_{sim}^{i} and Q_{obs}^{i} are the forecasted and observed *i*th value of the streamflow, $\bar{Q}_{obs}$ is the average of observed streamflow, N is the number of the observations.

3 Case study

3.1 Description of catchments and data

Huanren Reservoir and Xiangjiaba Hydropower Station are taken as case studies (see **Fig. 2**). The Huanren Reservoir Basin, located in the upper reaches of the Hunjiang River in China with a drainage area of 10400 km^2, is characterized by a mountainous environment with good vegetation. The annual average rainfall is about 860 mm with average annual streamflow of 142 m^3/s, 70% of which are from May-October. The Xiangjiaba Hydropower Station, located on the Jinshajiang River, is China's third-biggest hydropower station, following the Three Gorges Hydropower Station and Xiluodu Hydropower Station. The annual average streamflow in the river is about 3810 m^3/s and the annual average rainfall of the whole basin is about 756.4mm, 90% of which concentrates in the summer season.

The ACFs data, including 66 variables monitored by 160 nationwide stations is provided by the National Climate Center, China. Together with rainfall and streamflow, a total of 68 kinds of variables are taken as candidate inputs. The dataset for Huanren covers 600 months (1967-2016) and is partitioned into three parts, which are the training dataset (1967-1997), the cross-validation dataset (1998-2005), and the testing dataset (2006-2016). The dataset for Xiangjiaba covers a period of 528 months from 1967 to 2010, among which, 1967 to 1993, 1994 to 2000, and 2001 to 2010 are partitioned as

the training, cross-validation, and testing period, respectively.

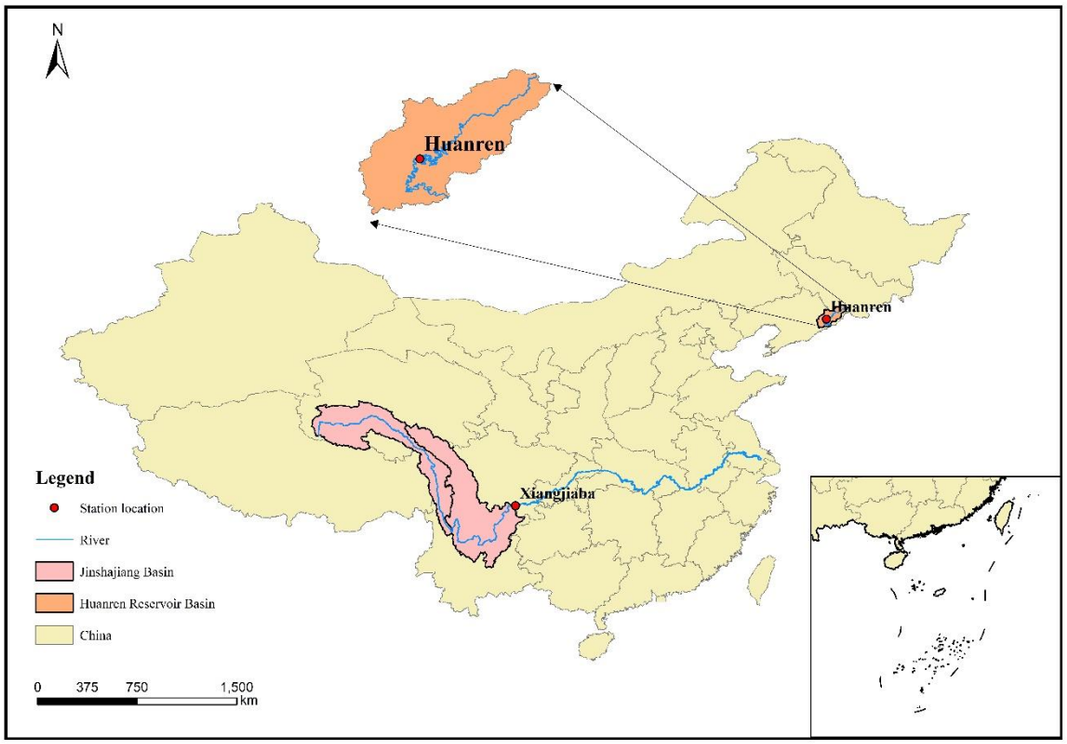

Fig. 2 Locations of Xiangjiaba and Huanren

3.2 Model development

In this subsection, the CNN, ANN, and ELM models are established to forecast the monthly streamflow in flood season (from May to October) for Huanren and the monthly streamflow in the whole year for Xiangjiaba.

(1) CNN model

Considering the relatively strong capability of CNN in data processing, 68 variables during the past 36 months are employed as predictors, and thus the size of the input layer is 36×68 pixels. The structure of CNN used in this study includes two convolution layers, two pooling layers, and a fully connected layer. Each convolution layer has only 1 channel and the hyper-parameters including learning rate, batch size, and the number of epochs are set as 1.0, 3, and 80, respectively. The kernel size in convolution and pooling layers are determined by trial and error, shown as follows:

a. Generate parameter combinations of model structure, which include kernel size in the two convolution layers and

pooling layers. The parameter combinations are sampled with kernel size in the convolution layer and pooling layer ranging from 1-20 and 1-8, respectively, and there are 25600 sets in total.

b. Exclude the infeasible parameter combinations, of which the dimension of the output matrix in the convolution layer or pooling layer violates the integer constraint.

c. Model training with feasible parameter combinations.

d. Eliminate relatively poor parameter combinations. When different parameter combinations lead to the same dense vector dimension, we only reserve the parameter combination that achieves the smallest *MAE* in the training period.

In order to avoid over-fitting in CNN (also do the same for the ANN and ELM), the training process is subjective to the stopping criteria where the cross-validation error increases for a specified number of iterations or the training reaches the scheduled maximum iterations. After the operation from step a to d, there are still many feasible structures. Among these different feasible structures, 9 parameter combinations, representing different structures listed in Table 1 were finally chosen for forecasting.

Table 1 Structure parameters of CNN models selected for monthly streamflow forecasting

Structure No.	Size of filters				The dimension of dense vector in the fully connected layer
	Convolution layer 1	Pooling layer 1	Convolution layer 2	Pooling layer 2	
1	13	4	5	2	5
2	13	4	3	2	12
3	17	1	19	2	17
4	7	2	12	2	20
5	15	1	19	2	36
6	5	2	14	1	57
7	13	4	2	1	65
8	7	2	12	1	80
9	19	1	16	1	105

（2）ANN and ELM models

For the ANN and ELM, their inputs are determined with CC and MI method, and four kinds of comparative models are established, i.e., CC-ANN, CC-ELM, MI-ANN, and MI-ELM. According to the previous introduction, we notice that the convolution and pooling operation in the CNN is a feature extraction process, which is equivalent to the input selection

process for the ANN and ELM. From this view, the dense vector can be considered as a selected input combination for the forecasting part in a CNN. Hence, the number of inputs for the ANN and ELM is designed to be consistent with the dimension of dense vectors in fully connected layers listed in Table 1. The input combinations for the ANN and ELM are listed in Table 2.

To identify the best architecture for the ANN, we follow the approach in (Zhang et al. 2015). A three-layer BP network model structure is used and the optimal number of neurons in the hidden layer is determined by using a heuristic method. Specifically, different numbers of neurons from 1 to 10 are tried 20 times and the architecture that acquires the smallest value of the *MAE* in the training period is taken as the optimal model. For the ELM, different numbers of hidden neurons (from 1 to 30) and different types of activation functions (sig, sine, hardline, tribas, radbas) are trained 20 times, and the model that achieves the smallest value of the *MAE* in the training period is determined as the optimal model.

Table 2 Input combinations for ANN and ELM

Method of selection	input combination	Number of the input variables	Method of selection	input combination	Number of the input variables
CC	C5	5	MI	M5	5
	C12	12		M12	12
	C17	17		M17	17
	C20	20		M20	20
	C36	36		M36	36
	C57	57		M57	57
	C65	65		M65	65
	C80	80		M80	80
	C105	105		M105	105

4 Results and discussions

The one-month-ahead streamflow forecast performance in the testing period of the three models for Huanren and Xiangjiaba are listed in Tables 3 and 4, respectively. As can be seen, CNN outperforms ANN and ELM, achieving higher *NSE*, lower *MAE*, and *RMSE* when the number of inputs is equal, and ELM presents a better performance in comparison with ANN in most situations. Taking the Xiangjiaba as an example, the error is reduced by 15.7 % and 25.5 % in *MAE*,

19.7 % and 37.9 % in *RMSE* from CNN to ELM and ANN for input combination C5 ($MAE_{\mathrm{CNN}}=891.21\ m^3/s$, $MAE_{\mathrm{ELM}}=1030.85\ m^3/s$, $MAE_{\mathrm{ANN}}=1118.96 m^3/s$, $RMSE_{\mathrm{CNN}}=1386.78\ m^3/s$, $RMSE_{\mathrm{ELM}}=1659.43\ m^3/s$, $RMSE_{\mathrm{ANN}}=1913.02 m^3/s$). When using the M20 input combination, a similar conclusion can be drawn ($MAE_{\mathrm{CNN}}=904.52\ m^3/s$, $MAE_{\mathrm{ELM}}=962.03\ m^3/s$, $MAE_{\mathrm{ANN}}=1152.26 m^3/s$, $RMSE_{\mathrm{CNN}}=1375.02\ m^3/s$, $RMSE_{\mathrm{ELM}}=1555.57\ m^3/s$, $RMSE_{\mathrm{ANN}}=1940.13 m^3/s$). The results also show that there is no strict increasing or decreasing trend for each performance index for any model, which implies that the inclusion of more inputs does not necessarily lead to better forecasts. Moreover, it can be noted that the forecasting for Xiangjiaba is more accurate than that for Huanren with *NSE* varies in the ranges of 0.13-0.84 and -0.55-0.37, respectively.

Table 3 Statistical indices for forecasts of Huanren by the CNN, ANN, and ELM in the testing period

Evaluation metric	Model	Number of inputs								
		5	12	17	20	36	57	65	80	105
MAE	CNN	119.02	122.02	110.75	108.80	111.83	117.99	117.73	123.21	129.28
	CC-ANN	138.85	157.77	137.31	141.33	160.44	172.59	186.60	194.20	194.77
	CC-ELM	145.69	132.20	122.87	132.62	129.17	130.96	143.59	136.88	133.77
	MI-ANN	125.88	160.57	183.69	157.55	188.56	137.84	191.13	179.43	176.60
	MI-ELM	120.79	146.48	144.22	135.66	138.60	128.37	140.94	133.94	135.25
RMSE	CNN	201.05	207.57	195.82	196.25	211.01	188.94	208.66	214.27	215.93
	CC-ANN	203.18	246.93	210.73	212.04	253.42	295.83	292.43	258.29	257.66
	CC-ELM	223.54	214.19	208.60	215.52	216.30	216.01	211.34	223.48	216.62
	MI-ANN	210.87	291.60	290.01	261.07	285.76	227.34	282.06	255.87	249.40
	MI-ELM	211.14	237.67	237.76	241.84	263.58	229.79	231.81	231.93	220.13
NSE	CNN	0.28	0.24	0.32	0.32	0.21	0.37	0.23	0.19	0.17
	CC-ANN	0.27	-0.08	0.21	0.20	-0.14	-0.55	-0.52	-0.18	-0.18
	CC-ELM	0.11	0.19	0.33	0.18	0.17	0.17	0.21	0.11	0.17
	MI-ANN	0.21	-0.51	-0.49	-0.21	-0.45	0.08	-0.41	-0.16	-0.10
	MI-ELM	0.21	0.00	0.00	-0.04	-0.23	0.06	0.05	0.05	0.14

Table 4 Statistical indices for forecasts of Xiangjiaba by the CNN, ANN, and ELM in the testing period

Evaluation metric	Model	Number of inputs								
		5	12	17	20	36	57	65	80	105
MAE	CNN	891.21	889.42	947.59	904.52	945.90	947.38	952.75	915.25	988.88
	CC-ANN	1118.96	1072.87	1035.89	1180.75	1207.65	1521.14	2058.99	2049.68	2037.72

	CC-ELM	1030.85	976.47	1016.09	956.84	1042.07	1025.97	1099.43	1119.85	1114.16
	MI-ANN	1189.31	1043.08	1718.60	1152.26	1553.53	1887.29	1671.96	1812.35	2138.22
	MI-ELM	932.70	1052.34	983.96	962.03	1013.46	1002.38	1136.28	1080.84	1179.70
RMSE	CNN	1386.78	1454.04	1521.66	1375.02	1455.98	1428.12	1437.22	1406.16	1470.65
	CC-ANN	1913.02	1842.85	1671.74	2006.48	2106.44	2361.55	3391.41	3043.03	2974.89
	CC-ELM	1659.43	1687.11	1518.36	1504.38	1708.49	1527.15	1656.86	1632.24	1706.22
	MI-ANN	1917.73	1813.27	3112.44	1940.13	2622.58	2962.95	2817.73	3070.36	3251.38
	MI-ELM	1727.11	1861.02	1626.48	1555.57	1579.54	1616.53	1689.03	1631.86	1750.68
NSE	CNN	0.84	0.83	0.81	0.84	0.83	0.83	0.83	0.84	0.82
	CC-ANN	0.70	0.72	0.77	0.67	0.64	0.54	0.06	0.24	0.27
	CC-ELM	0.77	0.77	0.81	0.81	0.76	0.81	0.77	0.78	0.76
	MI-ANN	0.70	0.73	0.20	0.69	0.43	0.28	0.35	0.23	0.13
	MI-ELM	0.76	0.72	0.78	0.80	0.80	0.79	0.77	0.78	0.75

Fig. 3 shows the detail of the best forecast results from each model in the testing period for Xiangjiaba, including the observed and forecasted data, absolute errors (forecasted value minus observed value), and relative errors (RE). As can be seen, the forecasts from the three models all fit well with the observations with an average relative error smaller than 10%. But the CNN underestimates the peaks of streamflow, while the ANN and ELM are more likely to overestimate the peaks. For medium flow, the forecast results of each model are close and all fit well with the observations. In terms of low flow, all the models underestimate the streamflow.

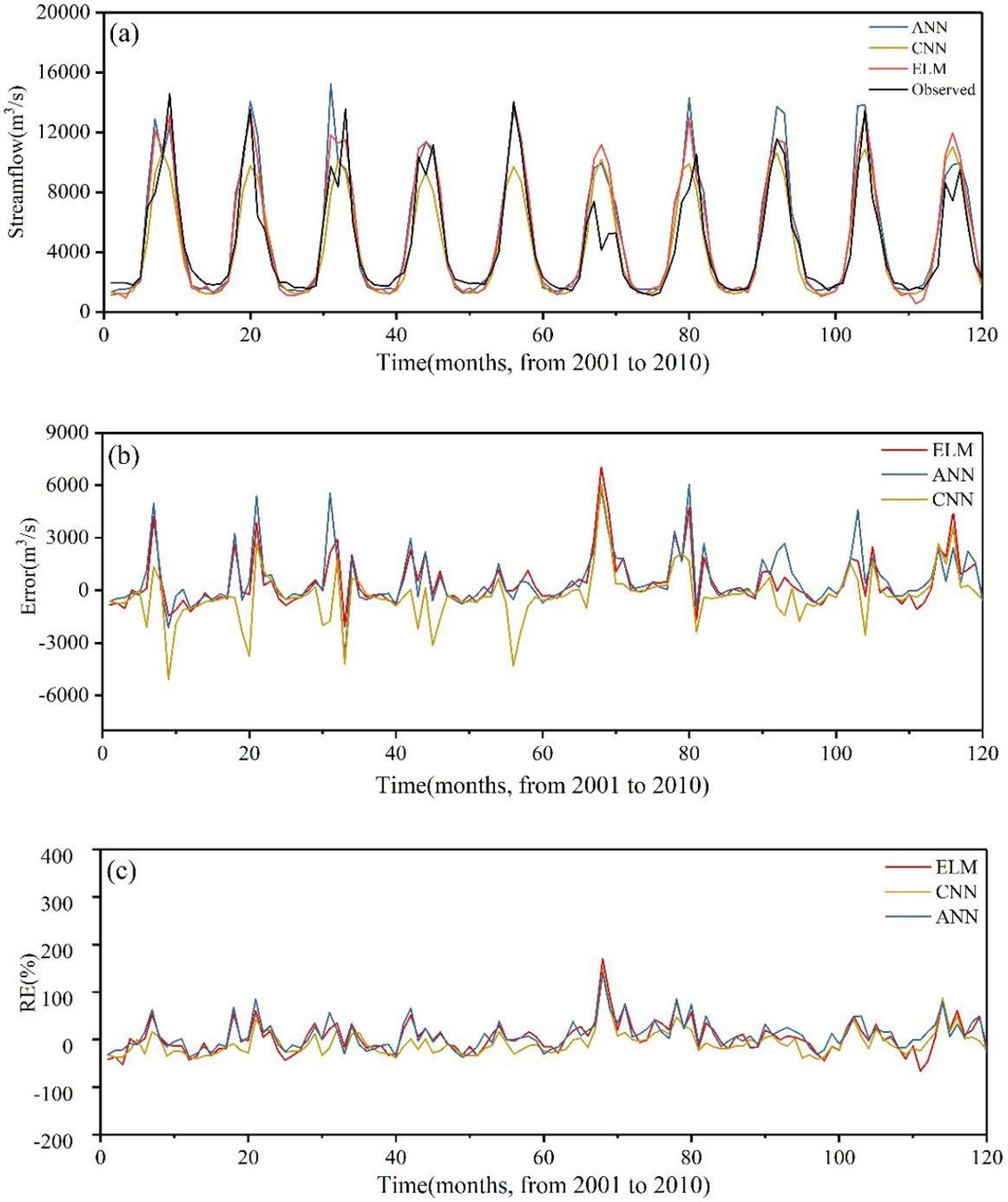

Fig. 3 The best forecast results achieved from CNN, ANN, and ELM for one-month-ahead streamflow on Xiangjiaba: (a) forecasted and observed value, (b) absolute errors, and (c) RE

Fig. 4 illustrates the best forecast results from each model in the testing period for Huanren. As shown in the figure, the peaks of streamflow in this region are mostly overestimated by each model except for some extremely high peaks. For the medium and low flow, the average mean of REs for each model is about 50% and the maximum RE is up to 500%.

Compared with the Xiangjiaba, the forecast performances by each model in Huanren are relatively poor, which might be attributed to the characteristics of the Huanren Reservoir basin. The Huanren is located in northern China, where the formation of rainfall in the flood season is strongly influenced by the mutual effects of Siberian cold air and the summer monsoon. The interaction between the Siberian cold air and the summer monsoon is very complicated and varies a lot, and thus leads to the difficult forecasting of the long-term streamflow. Besides, compared with Xiangjiaba, the streamflow in Huanren has a greater variation, which may also increase the difficulty in forecasts.

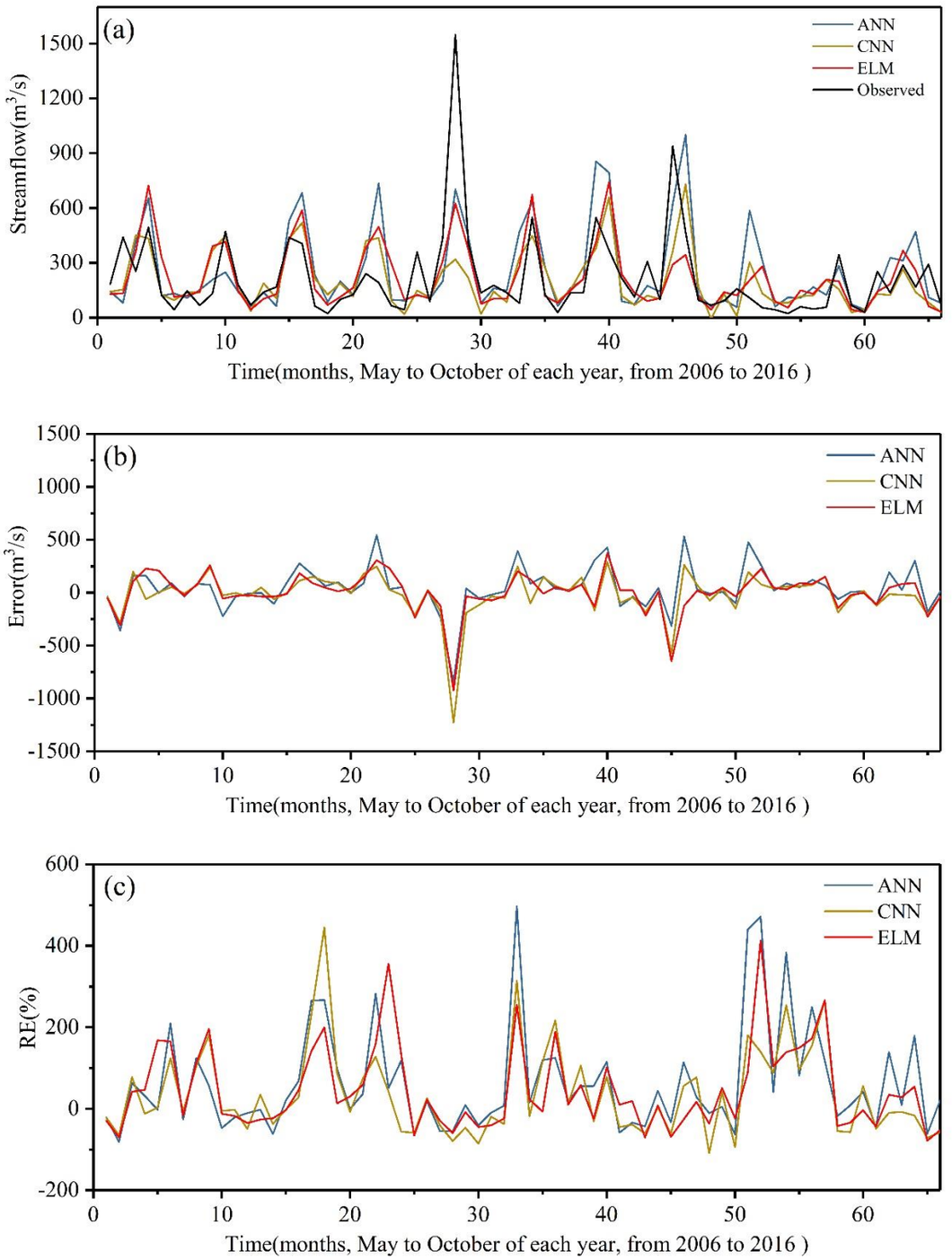

Fig. 4 The best forecast results achieved from CNN, ANN, and ELM for one-month-ahead streamflow in Huanren: (a) forecasted and observed value, (b) absolute errors, and (c) RE

The scatter-plots as well as the linear correlation coefficients (*r*) for the best forecast results from each model are shown in **Fig. 5**. As shown in **Fig. 5** (1), the data points of each model are scattered, which confirms the conclusion that the

forecast results for Huanren are relatively poor. Nevertheless, the CNN is still slightly better than the other two models with a relatively high *r*-value of 0.615. In **Fig. 5** (2), the data points of each model are concentrated in the vicinity of the diagonal $y = x$ and the correlation coefficients of all the three models are above 0.9. Although the *r*-value of the CNN is the lowest ($r_{\mathrm{CNN}} = 0.910$, $r_{\mathrm{ANN}} = 0.929$, $r_{\mathrm{ELM}} = 0.938$), it is only a 2.04% and 2.99% reduction compared with the ANN and ELM, respectively. Table 5 presents the metrics improvement of the CNN compared with the other two models in terms of their best forecast results. In Table 5, the metrics of the CNN are all better than that of ANN and ELM for the Huanren. For the Xiangjiaba, all of the metrics except for *r* outperform that of ANN and ELM.

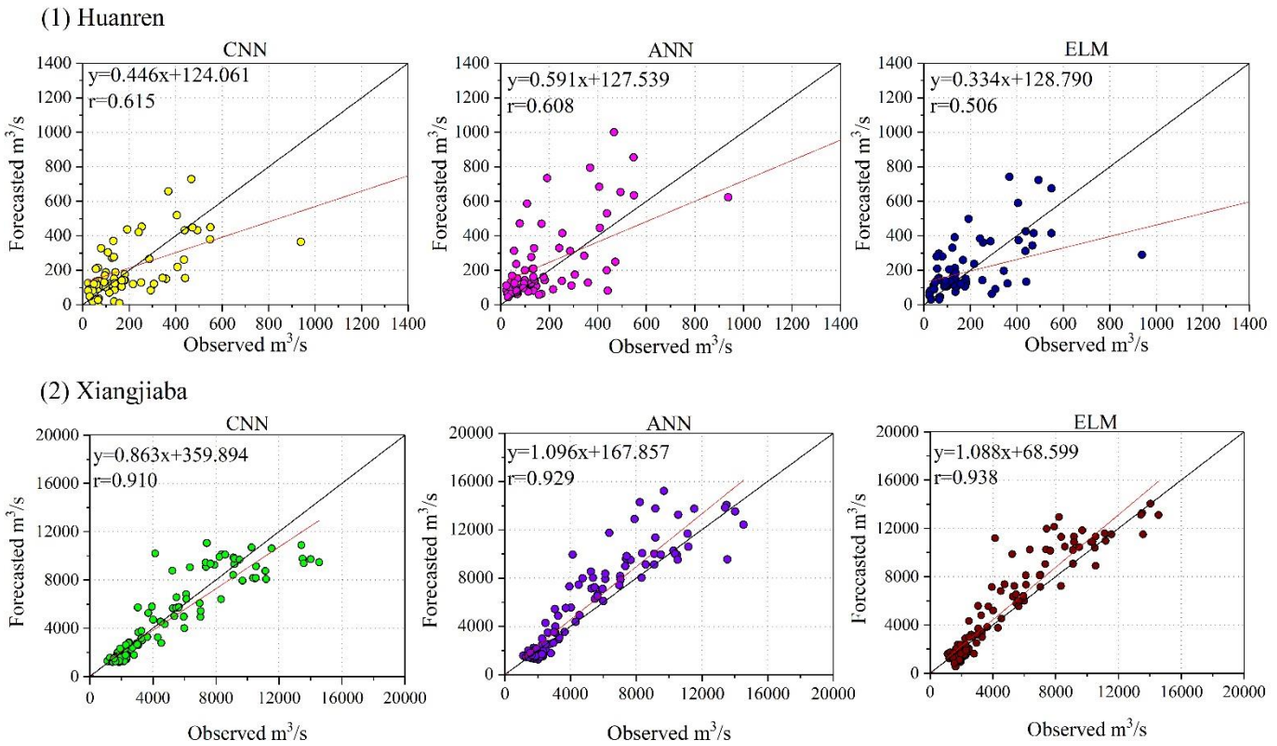

Fig. 5 Scatter-plots of observed and forecasted values within the testing period from optimal CNN, ANN, and ELM for Huanren and Xiangjiaba

Table 5 The performance metrics improvement achieved by CNN comparing the optimal forecast results among each model (%)

Station	Compared model	*MAE*	*RMSE*	*NSE*	*r*
Huanren	ELM	3.97	9.42	12.12	21.54
	ANN	14.07	10.34	76.19	1.15
Xiangjiaba	ELM	5.47	8.60	3.75	-2.99
	ANN	12.68	17.75	9.62	-2.05

In the process of establishing forecast models, it is not easy to catch the optimal input combination for a specific model. Therefore, the sensitivity of the model performance to input combinations is also a key indicator and worthy of attention. A lower sensitivity can make forecasts closer to the optimal value although the optimal input combination is not employed. **Fig. 6** displays the overall distribution of the performance metrics. It should be noted that for the metrics of *MAE* and *RMSE*, smaller median and shorter box height represent higher accuracy and stability, while for the metric of *NSE*, a larger median and shorter box height stand for more satisfying forecasts and lower sensitivity. For the Huanren, it can be seen that the ANN has a high sensitivity to the input combinations with a large box height of each performance metric. The CNN and ELM have nearly the same but better stability than the ANN. In comparison with the ELM, the CNN shows better accuracy because of its smaller box median of *MAE* and *RMSE*, and larger median of *NSE*. For the Xiangjiaba, a similar result can be seen.

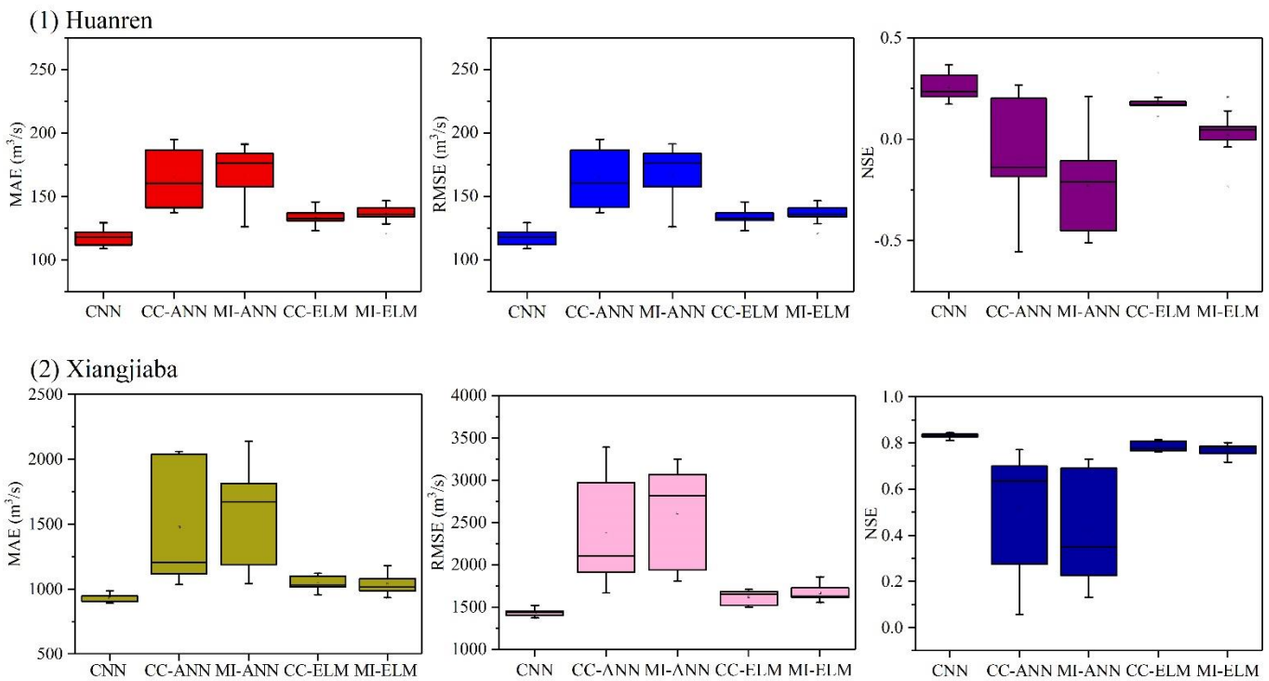

Fig. 6 Box-plots of the performance metrics from forecasted results of CNN, CC-ANN, MI-ANN, CC-ELM, and MI-ELM for Huanren and Xiangjiaba

From the results of applications in Huanren and Xiangjiaba, it can be seen that CNN outperforms its comparative

313 models with better forecast results and stable forecasting ability. Therefore, we can assert that CNN is an effective model
314 and can be used for monthly streamflow forecasting.

315 5 Conclusions

316 An attempt is made in this paper to explore the potential of the CNN for monthly streamflow forecasting. The monthly
317 streamflow and rainfall data from Huanren Reservoir and Xiangjiaba Hydropower Station and ACFs data from the National
318 Climate Center, China are employed for model training, validation, and testing. Nine cases of different structures of CNN
319 are designed with ACFs, rainfall and streamflow delayed for 1 to 36 months as inputs. The ANN and ELM are employed
320 as benchmarking yardsticks.

321 The results demonstrate that the CNN can be successfully applied to forecast the monthly streamflow. Comparing the
322 results of CNN, ANN, and ELM, it is seen that the CNN outperforms the ANN and ELM with smaller values of *MAE* and
323 *RMSE*, and higher values of *NSE*. With the change of model structure or input combinations, the CNN shows better stability
324 in forecasting accuracy than the ANN and ELM. The results also demonstrate that CNN, ANN, and ELM give satisfactory
325 forecasts for Xiangjiaba but are unable to maintain their accuracy for Huanren. Overall, the results and analysis presented
326 in this study demonstrate that the CNN is a superior and an alternative to the ANN and ELM in monthly streamflow
327 forecasting.

328 Although the results presented here are encouraging and the potential of the CNN for monthly streamflow forecasting
329 has been demonstrated, the CNN tends to underestimate or overestimate the peaks of streamflow and give relatively poor
330 forecasting for the streamflow with greater irregularity in some area. More case studies should be carried out to verify the
331 validity of the CNN for monthly streamflow forecasting. Besides, the implementation of feature extraction in CNN is a
332 two-dimensional scanning process in which multiple variables are involved in calculation simultaneously. This could raise
333 another drawback that lacks the capability of identifying specific variables dominating the variability of streamflow at a

specific location and requires further research.

Ethical Approval

Not applicable.

Consent to Participate

Not applicable.

Consent to Publish

Not applicable.

Authors' contributions

Yong Peng designed the study. Xingsheng Shu performed the research and wrote the initial draft of manuscript. Wei Ding analyzed the data and made revisions to the draft. Ziru Wang contributed to the revisions. Jian Wu contributed to the revisions. Min Li contributed to the revisions.

Funding

This work was supported by the National Key Research and Development Program of China [2016YFC0400906, 2017YFC0406005]; and the National Natural Science Foundation of China [91547111].

Competing Interest

Authors declare no conflict of interest.

Availability of data and materials

Not applicable.

References:

Abrahart RJ, See L. (2000). Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. HYDROL PROCESS, 14(11 - 12), 2157-2172. doi: 10.1002/1099-1085(20000815/30)14:11/12<2157::AID-HYP57>3.0.CO;2-S

Adamowski J, Sun K. (2010). Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds. J HYDROL, 390(1-2), 85-91. doi: 10.1016/j.jhydrol.2010.06.033

Afan HA, Allawi MF, El-Shafie A, Yaseen ZM, Ahmed AN, Malek MA, Koting SB, Salih SQ, Mohtar W, Lai SH, Sefelnasr A, Sherif M, El-Shafie A. (2020). Input attributes optimization using the feasibility of genetic nature inspired algorithm: Application of river flow forecasting. Sci Rep, 10(1), 4684. doi: 10.1038/s41598-020-61355-x

Babel MS, Badgujar GB, Shinde VR. (2015). Using the mutual information technique to select explanatory variables in artificial neural networks for rainfall forecasting. METEOROL APPL, 22(3), 610-616. doi: 10.1002/met.1495

Chong KL, Lai SH, Yao Y, Ahmed AN, Jaafar WZW, El-Shafie A. (2020). Performance Enhancement Model for Rainfall Forecasting Utilizing Integrated Wavelet-Convolutional Neural Network. WATER RESOUR MANAG, 34(8), 2371-2387. doi: 10.1007/s11269-020-02554-z

Cichocki A, Unbehauen R. (1993). *Neural Networks for Optimization and Signal Processing*. UK: Wiley-Teubner, Chichester.

Deo RC, Şahin M. (2016). An extreme learning machine model for the simulation of monthly mean streamflow water level in eastern Queensland. ENVIRON MONIT ASSESS, 188(2). doi: 10.1007/s10661-016-5094-9

Fu M, Fan T, Ding Z, Salih SQ, Al-Ansari N, Yaseen ZM. (2020). Deep Learning Data-Intelligence Model Based on Adjusted Forecasting Window Scale: Application in Daily Streamflow Simulation. IEEE ACCESS, 8; 8, 32632-32651. doi: 10.1109/ACCESS.2020.2974406

Ghorbani MA, Zadeh HA, Isazadeh M, Terzi O. (2016). A comparative study of artificial neural network (MLP, RBF) and support vector machine models for river flow prediction. ENVIRON EARTH SCI, 75(6). doi: 10.1007/s12665-015-5096-x

Hadi SJ, Tombul M, Salih SQ, Al-Ansari N, Yaseen ZM. (2020). The Capacity of the Hybridizing Wavelet Transformation Approach With Data-Driven Models for Modeling Monthly-Scale Streamflow. IEEE ACCESS, 8, 101993-102006. doi: 10.1109/ACCESS.2020.2998437

Haidar A, Verma B. (2018). Monthly Rainfall Forecasting Using One-Dimensional Deep Convolutional Neural Network. IEEE ACCESS, 6, 69053-69063. doi: 10.1109/ACCESS.2018.2880044

Huang C, Zhang J, Cao L, Wang L, Luo X, Wang J, Bensoussan A. (2020). Robust Forecasting of River-flow Based on Convolutional Neural Network. IEEE Transactions on Sustainable Computing, 1. doi: 10.1109/TSUSC.2020.2983097

Huang G, Zhu Q, Siew C. (2004-01-01). *Extreme learning machine: a new learning scheme of feedforward neural networks*, 2004. IEEE, p 985-990. doi: 10.1109/IJCNN.2004.1380068

Huang G, Zhu Q, Siew C. (2006). Extreme learning machine: Theory and applications. NEUROCOMPUTING, 70(1-3), 489-501. doi: 10.1016/j.neucom.2005.12.126

Hussain D, Hussain T, Khan AA, Naqvi SAA, Jamil A. (2020). A deep learning approach for hydrological time-series prediction: A case study of Gilgit river basin. EARTH SCI INFORM. doi: 10.1007/s12145-020-00477-2

Imrie CE, Durucan S, Korre A. (2000). River flow prediction using artificial neural networks: generalisation beyond the calibration range. J HYDROL, 233(1-4), 138-153. doi: 10.1016/S0022-1694(00)00228-6

Kuang D, Xu B. (2018). Predicting kinetic triplets using a 1d convolutional neural network. THERMOCHIM ACTA, 669, 8-15. doi: 10.1016/j.tca.2018.08.024

Li B, Cheng C. (2014). Monthly discharge forecasting using wavelet neural networks with extreme learning machine. Science China Technological Sciences, 57(12), 2441-2452. doi: 10.1007/s11431-014-5712-0

Li S, Goel L, Wang P. (2016). An ensemble approach for short-term load forecasting by extreme learning machine. APPL ENERG, 170, 22-29. doi: 10.1016/j.apenergy.2016.02.114

LIN J, CHENG C, CHAU K. (2006). Using support vector machines for long-term discharge prediction. Hydrological Sciences Journal, 51(4), 599-612. doi: 10.1623/hysj.51.4.599

Montanari A, Rosso R, Taqqu MS. (2000). A seasonal fractional ARIMA Model applied to the Nile River monthly flows at Aswan. WATER RESOUR RES, 36(5), 1249-1259. doi: 10.1029/2000wr900012

Mozo A, Ordozgoiti B, Gómez-Canaval S. (2018). Forecasting short-term data center network traffic load with convolutional neural networks. PLOS ONE, 13(2), e191939. doi: 10.1371/journal.pone.0191939

Ozgur Kisi MA. (2004). River flow modeling using artificial neural networks. J HYDROL ENG, 9(60), 60-63

Pashova L, Popova S. (2011). Daily sea level forecast at tide gauge Burgas, Bulgaria using artificial neural networks. J SEA RES, 66(2), 154-161. doi: 10.1016/j.seares.2011.05.012

Quilty J, Adamowski J, Khalil B, Rathinasamy M. (2016). Bootstrap rank-ordered conditional mutual information (broCMI): A nonlinear input variable selection method for water resources modeling. WATER RESOUR RES, 52(3), 2299-2326. doi: 10.1002/2015WR016959

Sahay RR, Srivastava A. (2014). Predicting Monsoon Floods in Rivers Embedding Wavelet Transform, Genetic Algorithm and Neural Network. WATER RESOUR MANAG, 28(2), 301-317. doi: 10.1007/s11269-013-0446-5

Samsudin R, Saad P, Shabri A. (2011). River flow time series using least squares support vector machines. HYDROL EARTH SYST SC, 15(6), 1835-1852. doi: 10.5194/hess-15-1835-2011

Shabri A, Suhartono. (2012). Streamflow forecasting using least-squares support vector machines. Hydrological sciences journal, 57(7), 1275-1293. doi: 10.1080/02626667.2012.714468

Siqueira H, Boccato L, Luna I, Attux R, Lyra C. (2018). Performance analysis of unorganized machines in streamflow forecasting of Brazilian plants. APPL SOFT COMPUT, 68, 494-506. doi: 10.1016/j.asoc.2018.04.007

Vapnik V. (1995). *The nature of Statistical Learning Theory*. Berlin: Spring Verlag.

Wu CL, Chau KW. (2006). *Evaluation of several algorithms in forecasting flood* (4013ed.): Adv Appl Artif Intell.

Yaseen ZM, Jaafar O, Deo RC, Kisi O, Adamowski J, Quilty J, El-Shafie A. (2016). Stream-flow forecasting using extreme learning machines: A case study in a semi-arid region in Iraq. J HYDROL, 542, 603-614. doi: 10.1016/j.jhydrol.2016.09.035

Yaseen ZM, Mohtar WHMW, Ameen AMS, Ebtehaj I, Razali SFM, Bonakdari H, Salih SQ, Al-Ansari N, Shahid S. (2019). Implementation of Univariate Paradigm for Streamflow Simulation Using Hybrid Data-Driven Model: Case Study in Tropical Region. IEEE ACCESS, 7, 74471-74481. doi: 10.1109/ACCESS.2019.2920916

Yaseen ZM, Sulaiman SO, Deo RC, Chau K. (2019). An enhanced extreme learning machine model for river flow forecasting: State-of-the-art, practical applications in water resource engineering area and future research direction. J HYDROL, 569, 387-408. doi: 10.1016/j.jhydrol.2018.11.069

Yılmaz I, Yuksek AG. (2008). An Example of Artificial Neural Network (ANN) Application for Indirect Estimation of Rock Parameters. ROCK MECH ROCK ENG, 41(5), 781-795. doi: 10.1007/s00603-007-0138-7

YU P, TSENG T. (1996). A model to forecast flow with uncertainty analysis. Hydrological Sciences Journal, 41(3), 327-344. doi: 10.1080/02626669609491506

Yu P, Yang T, Chen S, Kuo C, Tseng H. (2017). Comparison of random forests and support vector machine for real-time radar-derived rainfall forecasting. Journal of hydrology (Amsterdam), 552, 92-104. doi: 10.1016/j.jhydrol.2017.06.020

Zhang X, Peng Y, Zhang C, Wang B. (2015). Are hybrid models integrated with data preprocessing techniques suitable for monthly streamflow forecasting? Some experiment evidences. J HYDROL, 530, 137-152. doi: 10.1016/j.jhydrol.2015.09.047