Water Practice & Technology

Water Practice & Technology Vol 18 No 10, 2401 doi: 10.2166/wpt.2023.144

Deep and machine learning for daily streamflow estimation: a focus on LSTM, RFR and XGBoost

Özlem Terzi [a], Ecir Uğur Küçüksille [b], Tahsin Baykal [c] and Emine Dilek Taylan [c,*]

[a] Department of Civil Engineering, Isparta University of Applied Sciences, Isparta, Turkey
[b] Department of Computer Engineering, Suleyman Demirel University, Isparta, Turkey
[c] Department of Civil Engineering, Suleyman Demirel University, Isparta, Turkey
*Corresponding author. E-mail: dilektaylan@sdu.edu.tr

ÖT, 0000-0001-6429-5176; EUK, 0000-0002-3293-9878; TB, 0000-0001-6218-0826; EDT, 0000-0003-0734-1900

ABSTRACT

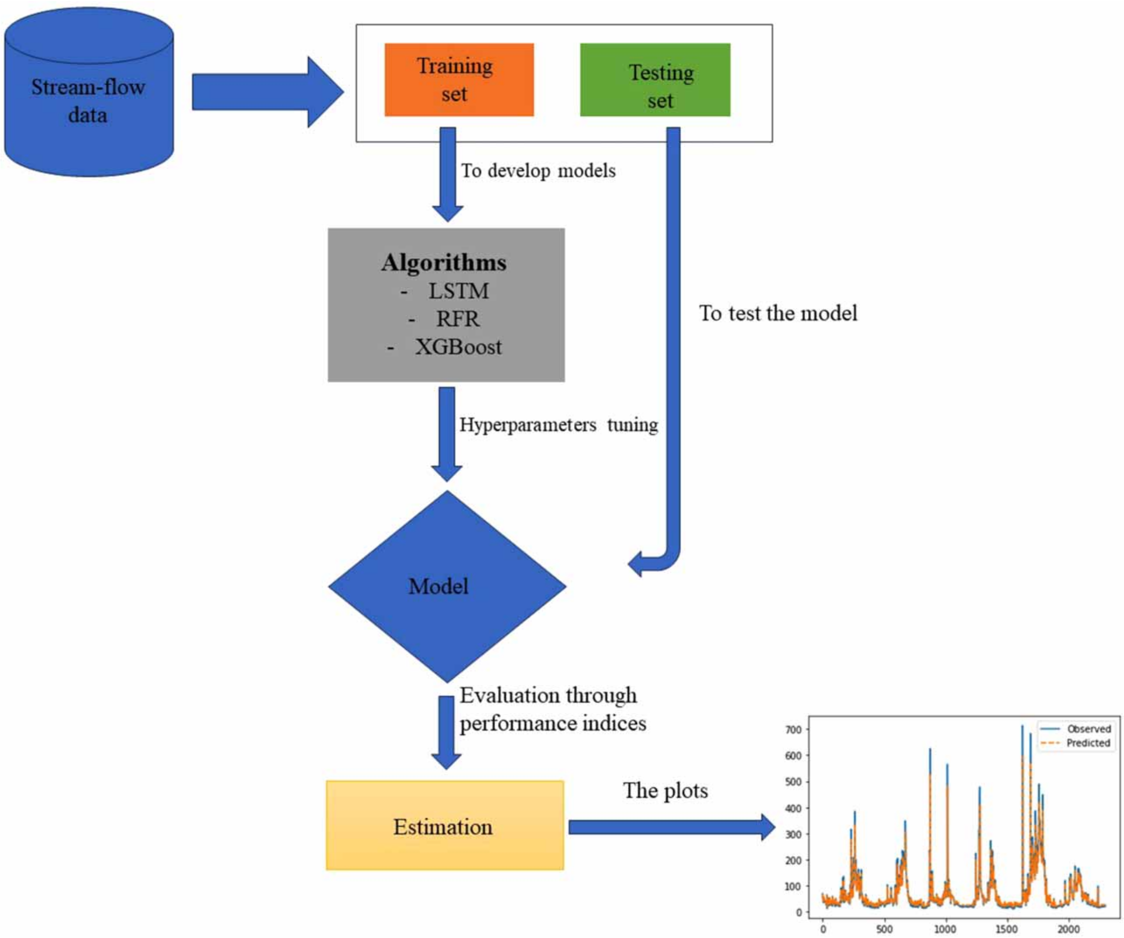
Estimation accuracy of streamflow values is of great importance in terms of long-term planning of water resources and taking measures against disasters such as drought and flood. The flow formed in a river basin has a complex physical structure that changes depending on the characteristics of the basin (such as topography and vegetation), meteorological factors (such as precipitation, evaporation and infiltration) and human activities. In recent years, deep and machine learning techniques have attracted attention thanks to their powerful learning capabilities and accurate and reliable modeling of these complex and non-linear processes. In this paper, long short-term memory (LSTM), random forest regression (RFR) and extreme gradient boosting (XGBoost) approaches were applied to estimate daily streamflow values of Göksu River, Turkey. Hyperparameter optimization was realized for deep and machine learning algorithms. The daily flow values between the years 1990–2010 were used and various input parameters were tried in the modeling. Examining the performance (R^2, RMSE and MAE) of the models, the XGBoost model having five input parameters provided more appropriate results than other models. The R^2 value of the XGBoost model was obtained as 0.871 for the testing set. Also, it is shown that deep and machine learning algorithms are used successfully for streamflow estimation.

Key words: extreme gradient boosting, Göksu River, LSTM, random forest regression, streamflow

HIGHLIGHTS

- Using deep and machine learning algorithms such as LSTM, RFR and XGBoost for streamflow estimation of Göksu River.
- Development of models for different input combinations.
- Hyperparameter tuning for the developed models and determining the best model structure.

GRAPHICAL ABSTRACT



1. INTRODUCTION

Flow estimation in the ungauged parts of the rivers is one of the frequently encountered problems for hydrologists. Making reliable flow estimations plays an important role in the planning of water resources. The reliable predictions can minimize the possible errors occurring in the studies like flood control and planning of hydraulic structures. However, it is not always possible to reach the required flow data at the desired places due to some reasons. Therefore, it has been necessary to develop various flow estimation methods. These methods can generally be classified as deterministic basin models (Box & Jenkins 1970; Salas 1980; Bender & Simonovic 1994; Modarres 2007; Valipour *et al.* 2013). However, these techniques are time-consuming and require too many parameters. Apart from these, there are artificial intelligence methods such as artificial neural networks (ANNs), which are often used in recent years and easier to use in flow prediction studies because they do not require too many parameters. In water resources studies, hydrologists frequently use ANNs. Indeed, ANNs have been used in forecasting rainfall (Zhang *et al.* 1997; Chiang *et al.* 2007), sediment (Partal & Cigizoglu 2008; Jothiprakash & Garg 2009), floods (Campolo *et al.* 2003; Chang *et al.* 2007; Aziz *et al.* 2015), evaporation (Moghaddamnia *et al.* 2009) and flow (Panagoulia 2006; Pramanik & Panda 2009; Kostić *et al.* 2016; Veintimilla-Reyesa *et al.* 2016; Zemzami & Benaabidate 2016). Uysal *et al.* (2016) developed snowmelt models with ANNs (multi-layer perceptron (MLP) and radial basis function) for mountainous region in Turkey. They showed that both models give similar results. Yin *et al.* (2016) compared the rotated general regression neural network, the general regression neural network, the feed-forward error back-propagation model and the soil moisture accounting and routing model to forecast monthly river flow in Heihe River, China. They stated that the performance of the rotated general regression neural network model was better than other models.

ANNs have been successful in many areas and problems. However, the hardware improvements are insufficient when the number of hidden layers and nodes is increased. Therefore, its use has come to a standstill due to hardware constraints. However, ANNs have switched from shallow networks to deep networks through GPU and other hardware improvements. In recent years, deep neural networks (DNNs) have attracted great attention by their capability of modeling the huge amount of data with multiple hidden layers, which makes them more advantageous than the traditional neural networks (Coulibaly *et al.* 2000; Zhang *et al.* 2001; Kentel 2009; Yaseen *et al.* 2016; Zhu *et al.* 2020; Mohammadi *et al.* 2021). Li *et al.* (2016) developed DNN models to forecast daily inflow for two reservoirs in China. They compared the models to the basic feed forward neural network and the autoregressive (AR) integrated moving average models. The results show that the DNN models are suitable for performance criteria. Bai *et al.* (2016) proposed a multiscale deep feature learning method for forecasting the daily inflow values of the Three Gorges Reservoir in China. They stated that the used method has good performance of forecasting peak values. Assem *et al.* (2017) aimed to provide DNNs model for predicting flow and water level of Shannon River, Ireland. They show that the model performs better than time-series prediction models. They stated that the proposed method could be very convenient for better planning of water resources. Tao *et al.* (2016) applied a DNNs framework for correcting the estimation bias to satellite-based precipitation estimation products. They indicated that DNNs could extract utility information for estimation of precipitation. Also, the methodology can help to find out further features from satellite datasets for reducing bias. Chen *et al.* (2012) developed the drought model using deep belief networks (DBNs). The drought index obtained from the standardized precipitation index is estimated with DBNs model for Huaihe River Basin in China. They noted that the model has better prediction performance than the back-propagation neural network.

Ensemble learning, one of the popular methods for various machine learning tasks, is frequently used in hydrology studies (Fan *et al.* 2019; Ma *et al.* 2021). Erdal & Karakurt (2013) investigated the use of two ensemble learning paradigms, namely bagging and stochastic gradient boosting in classification and regression trees (CART) in the field of flow prediction. They compared ensemble models with support vector machine (SVM) models. They found that the ensemble learning paradigms were successful in the training phase but did not achieve the same success in the testing phase. They combined bagging and stochastic gradient boosting paradigms to increase the success of the testing phase. As a result, they showed that the ensemble learning paradigms significantly increased success in CART algorithms. Galelli & Castelletti (2013) investigated the prediction ability of randomized trees in terms of accuracy, clarification ability and calculation efficiency in the flow modeling study. For this purpose, they identified Marina catchment (Singapore) and Canning River (Australia) as the study area. They showed that randomized trees had better performance compared to CART, M5, ANNs and multiple linear regressions. Zhao & Chen (2015) proposed a hybrid model based on ensemble empirical mode decomposition (EEMD) and AR for annual flow estimation. They tested the proposed model by using annual flow data obtained from four hydrological stations upstream of the Fenhe River Basin, China. Consequently, they found that the proposed hybrid model was more successful when they compared the EMD-AR and AR models with the proposed EEMD-AR. Zhang *et al.* (2018) suggested the hybrid EEMD-ENN model consisting of a combination of EEMD and Elman neural network (ENN) to overcome the difficulties of modeling and to increase the accuracy of prediction. In order to test this model, they used annual flow time series values obtained from four main streams in the Dongting Lake basin and four hydrological stations at the outlet of the lake. For this purpose, they have established four different models including back-propagation (BP) neural network, EEMD-BP, ENN and the proposed hybrid EEMD-ENN model. As a result, they have shown that the hybrid model provided better results. Nguyen (2015) used the random forest (RF) algorithm to estimate the incoming flow of Hoa Binh reservoir for 10 flow days. In conclusion, the developed model using the RF algorithm was suitable for predicting incoming flow values.

Extreme gradient boosting (XGBoost) is one of the ensemble learning algorithms. Compared to other algorithms, they offer more robust models with regular terms and column sampling (Chen & Guestrin 2016; Budholiya *et al.* 2022). This algorithm, which is used in different fields, has also become preferred in hydrology. Ni *et al.* (2020) created a hybrid model that uses a combination of XGBoost and Gaussian Mixture Model (GMM) for monthly streamflow estimation. They used monthly flow data from Cuntan and Hankou stations in the Yangtze River Basin to model. They proposed the model as a superior alternative for optimum management of water resources. Yu *et al.* (2020) made a 10-day flow forecast for the Three Gorges Dam in China with the FT-SVR (Fourier transform support vector regression) which was previously developed. The 10-day inflow time series was decomposed into seven components. They estimated each separated component with XGBoost and achieved better prediction results with FT-XGBoost. Using XGBoost, Venkatesan & Mahindrakar (2019) predicted 1–5 h ahead flooding for the Kolar Basin in India with hourly precipitation and runoff data from

1987 to 1989. The model results were compared with RF and SVM and it was stated that the XGBoost method performed better. Li *et al.* (2020) used elastic net regression (ENR), SVR, RF and XGBoost models for monthly flow forecast and a modified multi-model named modified stacking ensemble strategy (MSES) suggested as an integration method. They said that the RF and XGBoost models have better prediction performance than ENR and SVR. Vogeti *et al.* (2022) used bi-directional long short-term memory (Bi-LSTM), wavelet neural network (WNN) and XGBoost methods for flow estimation of the Lower Godavari Basin. 80% of 39 years of daily precipitation, evapotranspiration and flow data were reserved for model training and 20% for validation. XGBoost outperformed WNN and Bi-LSTM.

Long short-term memory (LSTM), which is frequently used by researchers in recent years (Katipoğlu & Sarıgöl 2023), was proposed by Hochreiter & Schmidhuber (1997). It can predict the periodic and complex structure of time series with high accuracy. Cheng *et al.* (2020) used ANN and LSTM models for flow prediction by using precipitation and flow data covering the 1974–2014 periods of Nan River Basin and Ping River Basin in Thailand. They stated that LSTM was better in daily flow prediction than ANN. Rahimzad *et al.* (2021) estimated daily flows with linear regression (LR), MLP, SVM and LSTM methods using 26 years of precipitation and flow data from the Kentucky River Basin in the USA. They said that LSTM performed better than others. Hu *et al.* (2020) used the flow data of a station in Tunxi, China and the precipitation data of 11 stations in the region and estimated the flow data at 6 h in the future with LSTM. They compared LSTM results with SVR and MLP. They stated that LSTM had a better performance with R^2 (0.97). Fu *et al.* (2020) used LSTM and classical back-propagation neural network model for flow prediction of the Kelantan River on the Malaysia Peninsula. They stated that the LSTM model better predicted rapidly changing flows during dry and rainy periods. Girihagama *et al.* (2022) used standard and attention-based encoder–decoder LSTM models to estimate flows from 10 different basins in the Ottawa River Basin in Canada. They observed that the encoder–decoder LSTM model had better performance in all basins.

This study addresses the follow aims: (i) it evaluates the performance of deep and machine learning methods in daily streamflow estimation, (ii) it identifies the optimal hyperparameters for daily streamflow estimation and (iii) it assesses the impact of various input combinations on daily streamflow estimation. For these purposes, daily streamflow values in the Göksu River, Turkey were used. The Karahacılı, Kırkkavak and Hamam stations located on the Göksu River were selected in developing models. In this context, deep learning (LSTM) and machine learning (RFR and XGBoost) algorithms were used as modeling approaches. Hyperparameter optimization was performed for LSTM, RFR and XGBoost algorithms using different input combinations and the most suitable model was selected. The rest of the article is structured as follows. In the second section, LSTM architecture and in the third section, ensemble learning algorithms (RFR and XGBoost) are given. In the fourth section, the study area and the data used in the models are introduced. In the fifth section, the selection of hyperparameters of the models and the model results are presented. In the sixth section, the results of the study are examined.

2. MATERIALS AND METHODS

2.1. Study region and data

The Göksu River, which starts from the Mediterranean Taurus Mountains in two tributaries, is the most important river of Mersin province in Turkey. The Göksu River, which is located within the borders of Konya, Karaman, Antalya and Mersin provinces, flows into the Mediterranean Sea. The Göksu River has northern and southern tributaries, namely Gökçay River and Gökdere River. After these two tributaries merge in the Mut district, its name becomes the Göksu River. The length of the river is about 260 km. The study area and the flow observation stations are given in Figure 1.

Flow data of three observation stations belong to Electrical Power Resources Survey and Development Administration is used in the Göksu River to develop LSTM, RFR and XGBoost models. The 7,665 daily flow values have been obtained for the Karahacılı (1714) on Gökçay River, Kırkkavak (1719) on Gökdere River and Hamam (1720) on Göksu River stations for the years 1990–2010.

2.2. Long short-term memory

Recurrent Neural Network (RNN) can be defined as a type of artificial neural network used to model time series data (Jordan 1986; Rumelhart *et al.* 1986). An RNN can process current data using previous data. However, RNN has the problem of training of long-term dependency data. LSTM was introduced by Hochreiter & Schmidhuber to overcome the limitations of RNN (Hochreiter & Schmidhuber 1997). Here, the hidden layer unit called memory cells is used. Memory cells have three gates that store the transient state of the network, called the

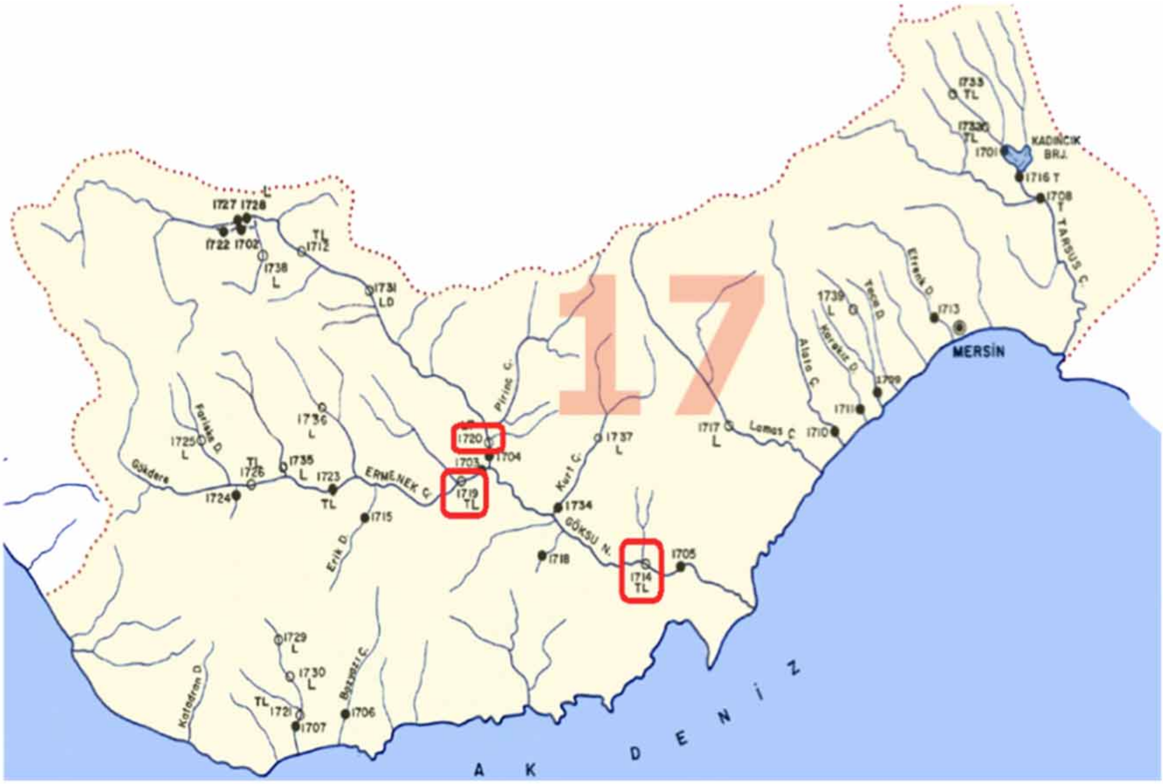


Figure 1 | The Göksu River Basin.

input gate, the output gate and the forget gate. The input and output gates are used to control the flow of memory cell inputs and outputs to the rest of the network. The forget gate is used to transmit the output information from the previous neuron to the next neuron with high weights. The information contained in memory depends on high activation results. That is, if the input unit has high activation, the information is stored in the memory cell. In addition, if the output unit has high activation, it transmits the information to the next neuron. Otherwise, the high-weighted input information is in the memory cell (Shahid *et al.* 2020). Figure 2 shows the internal structure of an LSTM cell.

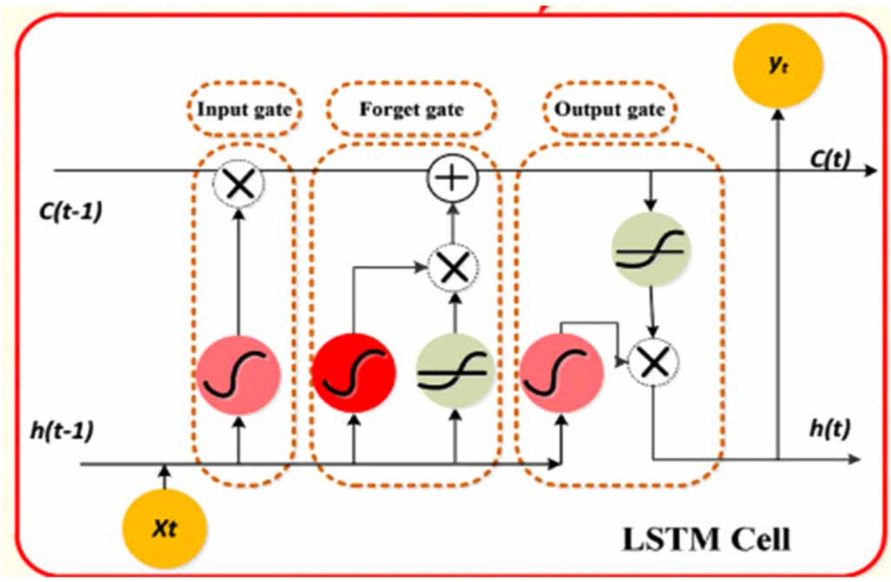


Figure 2 | Internal structure of an LSTM cell (Shahid *et al.* 2020).

2.3. Ensemble learning

The principal advantages of ensemble models over single models are described with their advanced generalization capabilities and the flexible functional mapping between the variables of the system. Ensemble learning usually consists of three phases, which are resampling, generation of sub-models and pruning and ensemble integration. Resampling, dealing with the generation of subsets of data from the original dataset, is usually the primary character behind a key-ensemble model. The generation of sub-models defines the process of selecting some suitable regression models for the system. To prune the sub-models, it determines the most suitable ensemble configurations and the structures of sub-models. Finally, ensemble integration is a special method, which transforms or chooses the estimates coming from the members, creating the ensemble estimate. In the literature, the most common ensemble learning frameworks are Bagging, Stacking and Boosting (Alobaidi *et al.* 2018).

The bagging ensemble, which is proposed by Breiman (1996), is one of the most effective techniques for increasing the predictive accuracy of individual classifiers and reducing the variance. In order to create multiple versions of classifiers, Bagging ensemble uses bootstrap samples created by sampling uniformly instances from training datasets. The final results of the different versions of the classifiers are combined by simple voting to obtain a general estimation. In case of regression, the average estimate is given as a result (Pham *et al.* 2017).

Stacking, which is the technique presented by Wolpert (1992), involves the creation of linear combinations of different estimators to obtain generally improved results. It consists of two stages. In the first stage, different models such as J48, Naïve Bayes and RF are learned according to the dataset and their output help create new datasets. In the second stage, the dataset is used with the learning algorithm to obtain the final output.

Boosting is another type of effective ensemble methodology proposed by Freund & Schapire (1997) based on the learning in sequence. Learning is primarily done on the completed dataset and then on the resultant data obtained from the last learning performance. The main point is to change the weights of each training data point, explicitly. The obtained weights are increased for the misclassified sample data points. In a similar way, they are decreased for the correctly classified sample data points (Verma & Mehta 2017).

2.3.1. Random forest regression

Random forest regression (RFR) is a tree-based technique containing the stratification or segmentation of the predicator space into a series of simple regions (Wu *et al.* 2017). In other words, it is a nonparametric ensemble learning method gathering the results obtained from many individual decision trees. Overfitting is prevented by growing each tree using a bootstrap sample and by selecting from a random subset of variables at each split. Observations, which are not involved in a sample of tree because of the bootstrapping procedure, are called out-of-bag (on average about 36%). They serve as the test set of the tree and they are used in measuring the prediction error. The importance of an interest predictor can be estimated with permutation, by randomly shuffling its values in the out-of-bag samples and comparing the final prediction error to the error obtained prior to the shuffle. The importance estimate formed in this way is called a VIMP and it includes the effects of all interactions, because it removes the effect of predictor on the choice of other variables deeper in the tree (Van der Meer *et al.* 2017).

2.3.2. Extreme gradient boosting

XGBoost is a machine learning algorithm proposed by Chen & Guestirn in 2016. Boosting Tree algorithms are based on a decision tree known as a classification and regression tree (CART) (Dong *et al.* 2020).

The XGBoost model uses the additive training method to optimize the objective function. This means that the optimization process of the second step depends on the result of the previous step. The advantage of XGBoost is that it supports linear classifiers and performs second-order Taylor expansion of the cost function to make the results more accurate. The score of the loss function used in the XGBoost algorithm and the solution of the weights can be expressed as follows:

$$w_j^* = -\frac{\sum g_i}{\sum h_i + \lambda} \tag{1}$$

$$obj^* = -\frac{1}{2}\sum_{j=1}^{T}\frac{(\sum g_i)^2}{\sum h_i + \lambda} + \gamma T \tag{2}$$

where obj^* represents the score of the loss function. The smaller the score, the better the structure of the tree. w_j^* represents the solution of weights (Jiang *et al.* 2020).

2.4. Hyperparameter tuning

Hyperparameters are parameters that are determined initially before the learning process starts in a machine learning model. The values of these parameters do not change when the learning process is over. Learning rate, epoch, etc., parameters can be given as examples.

Hyperparameter values given by default in machine learning models do not guarantee the best performance (Schratz *et al.* 2019). Therefore, tuning hyperparameter values can greatly affect the performance of the model (Mantovani *et al.* 2016). The large amount of hyperparameters of the models makes it almost impossible to manually adjust these values. For this reason, many hyperparameter tune methods are available and used in the literature. The most used methods are GridSearch and RandomSearch.

GridSearch creates a new model by trying all possible combinations from the given collection of values for each hyperparameter and returns the hyperparameter combination that provides the highest accuracy. The problem with this method is that the process takes a long time when there are too many hyperparameters and values to try. This causes the method to run very slowly.

In the RandomSearch method, N combinations determined from each hyperparameter value collection are randomly selected and return the hyperparameter combination that provides the highest accuracy. With this method, a search can be made much faster and with an accuracy close to grid search.

3. RESULTS AND DISCUSSION

The study is conducted for the Göksu River in Turkey, with the aim of improving the accuracy of streamflow value predictions for water resource planning and disaster mitigation against events such as droughts and floods. The complex nature of river basin flows, influenced by basin characteristics (e.g., topography, vegetation), meteorological factors (e.g., precipitation, evaporation, infiltration) and human activities, necessitates advanced modeling techniques. In recent years, deep and machine learning methods have garnered attention due to their ability to effectively model complex and nonlinear processes. Daily flow values of the Göksu River spanning the years 1990–2010 constitute the dataset for this study. This study focuses on estimating daily streamflow values using deep and machine learning techniques, specifically LSTM, RFR and XGBoost approaches. These methods are chosen for their capabilities in capturing temporal dependencies, handling nonlinear relationships and providing accurate predictions. Each model is trained on a portion of the dataset and hyperparameter optimization is performed to enhance their predictive performance. In the modeling stage, data of three flow observation stations on Göksu River was used. The input parameters are flow data of Kırkkavak and Hamam stations while the output parameter is flow data of Karahacılı station. The various models were developed with seven different input combinations. It was used the previous 1-day (Q_{t-1}), 2-day (Q_{t-2}), 3-day (Q_{t-3}), 4-day (Q_{t-4}) and 5-day (Q_{t-5}) flow values of Karahacılı station, and daily flow values of Kırkkavak ($Q_{(1719)t}$) and Hamam ($Q_{(1720)t}$) stations as input parameters in modeling. The total number of data used in the modeling stage is 7,662. In the models, 80% of the total data (1990–2005 years) is allocated to the training set and the remaining 20% of the total data (2005–2010 years) is used in the testing set. Then, the whole data are normalized from 0 to 1 with the help of MinMaxScaler class of sklearn library. In the LSTM, RFR and XGBoost models, python programming language, numpy (URL-1 2018a), pandas (URL-2 2018b), sklearn (URL-3 2018c) and keras (URL-4 2018d) libraries are used. For LSTM, RFR and XGBoost methods, the best model structure was determined belong to each input combination. The hyperparameters selected for the best models are summarized in Table 1.

As mentioned above, LSTM, RFR and XGBoost models have been developed by various input combinations. The performance of the models is assessed using key metrics such as root mean squared error (RMSE), mean absolute error (MAE) and coefficient of determination (R^2). These metrics provide insights into the accuracy and reliability of the models' estimations. RMSE, MAE and R^2 values of LSTM, RFR and XGBoost models are given in Tables 2–4, respectively. Examining LSTM models in Table 2, it was shown that the higher R^2 values than 0.8 in developed models are obtained except model 6. Model 1, having Q_{t-1} parameter, has the highest R^2 value. The RMSE, MAE and R^2 values of model 1 are 28.68 m^3/s, 12.04 m^3/s and 0.853 for the testing set, respectively.

Table 1 | Hyperparameters tuning

Models	Hyperparameters	Selection
LSTM	Neurons in the input layer	70
	Numbers of hidden layers	3
	Neurons in each hidden layer	70
	Neurons in the output layer	One neuron
	Activation function	relu
	Number of epochs	100
	Batch size	32
	Dropout	0.2
	Loss function optimizer	Mean Squared Error
RFR	n_estimators	200
	min_samples_split	5
	min_samples_leaf	4
	max_features	Auto
	max_depth	10
	bootstrap	True
XGBoost	subsample	0.8
	reg_lambda	2
	reg_alpha	1
	n_estimators	300
	min_child_weight	19
	max_depth	6
	learning_rate	0.1
	gamma	0
	colsample_bytree	0.8

Table 2 | RMSE, MAE and R^2 values of the LSTM models

		Training set			Testing set		
Model no	Input parameters	RMSE (m^3/s)	MAE (m^3/s)	R^2	RMSE (m^3/s)	MAE (m^3/s)	R^2
1	Q_{t-1}	25.16	10.27	0.852	28.68	12.04	0.853
2	Q_{t-1}, Q_{t-2}	25.48	10.15	0.850	29.20	11.91	0.850
3	$Q_{t-1}, Q_{t-2}, Q_{t-3}$	26.00	10.31	0.837	30.06	12.21	0.833
4	$Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}$	26.99	10.27	0.823	31.59	12.45	0.815
5	$Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}, Q_{t-5}$	27.16	10.24	0.820	31.79	12.54	0.813
6	$Q_{(1719)t}, Q_{(1720)t}$	41.96	19.89	0.581	50.41	28.36	0.545
7	$Q_{(1719)t}, Q_{(1720)t}, Q_{t-1}$	25.23	9.74	0.848	28.37	12.12	0.850

Table 3 | RMSE, MAE and R^2 values of the RFR models

		Training set			Testing set		
Model no	Input parameters	RMSE (m^3/s)	MAE (m^3/s)	R^2	RMSE (m^3/s)	MAE (m^3/s)	R^2
1	Q_{t-1}	21.80	8.53	0.884	27.69	11.59	0.858
2	Q_{t-1}, Q_{t-2}	19.43	7.09	0.908	29.41	12.59	0.866
3	$Q_{t-1}, Q_{t-2}, Q_{t-3}$	18.93	6.69	0.912	26.45	11.24	0.870
4	$Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}$	18.25	6.28	0.919	26.52	11.33	0.869
5	$Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}, Q_{t-5}$	18.37	6.74	0.918	26.40	11.15	0.871
6	$Q_{(1719)t}, Q_{(1720)t}$	33.85	16.15	0.723	53.83	34.08	0.528
7	$Q_{(1719)t}, Q_{(1720)t}, Q_{t-1}$	19.48	6.95	0.907	27.45	11.48	0.860

Table 4 | RMSE, MAE and R^2 values of the XGBoost models

		Training set			Testing set		
Model no	**Input parameters**	**RMSE (m^3/s)**	**MAE (m^3/s)**	R^2	**RMSE (m^3/s)**	**MAE (m^3/s)**	R^2
1	Q_{t-1}	23.52	8.72	0.865	27.03	10.89	0.865
2	Q_{t-1}, Q_{t-2}	20.94	8.42	0.893	27.58	11.66	0.859
3	Q_{t-1}, Q_{t-2}, Q_{t-3}	21.43	8.23	0.888	26.63	10.96	0.868
4	Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}	20.55	7.99	0.897	26.42	11.07	0.870
5	Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}, Q_{t-5}	20.47	8.02	0.898	26.38	11.04	0.871
6	$Q_{(1719)t}$, $Q_{(1720)t}$	39.15	18.88	0.626	52.86	32.66	0.538
7	$Q_{(1719)t}$, $Q_{(1720)t}$, Q_{t-1}	21.76	8.61	0.884	27.61	11.62	0.859

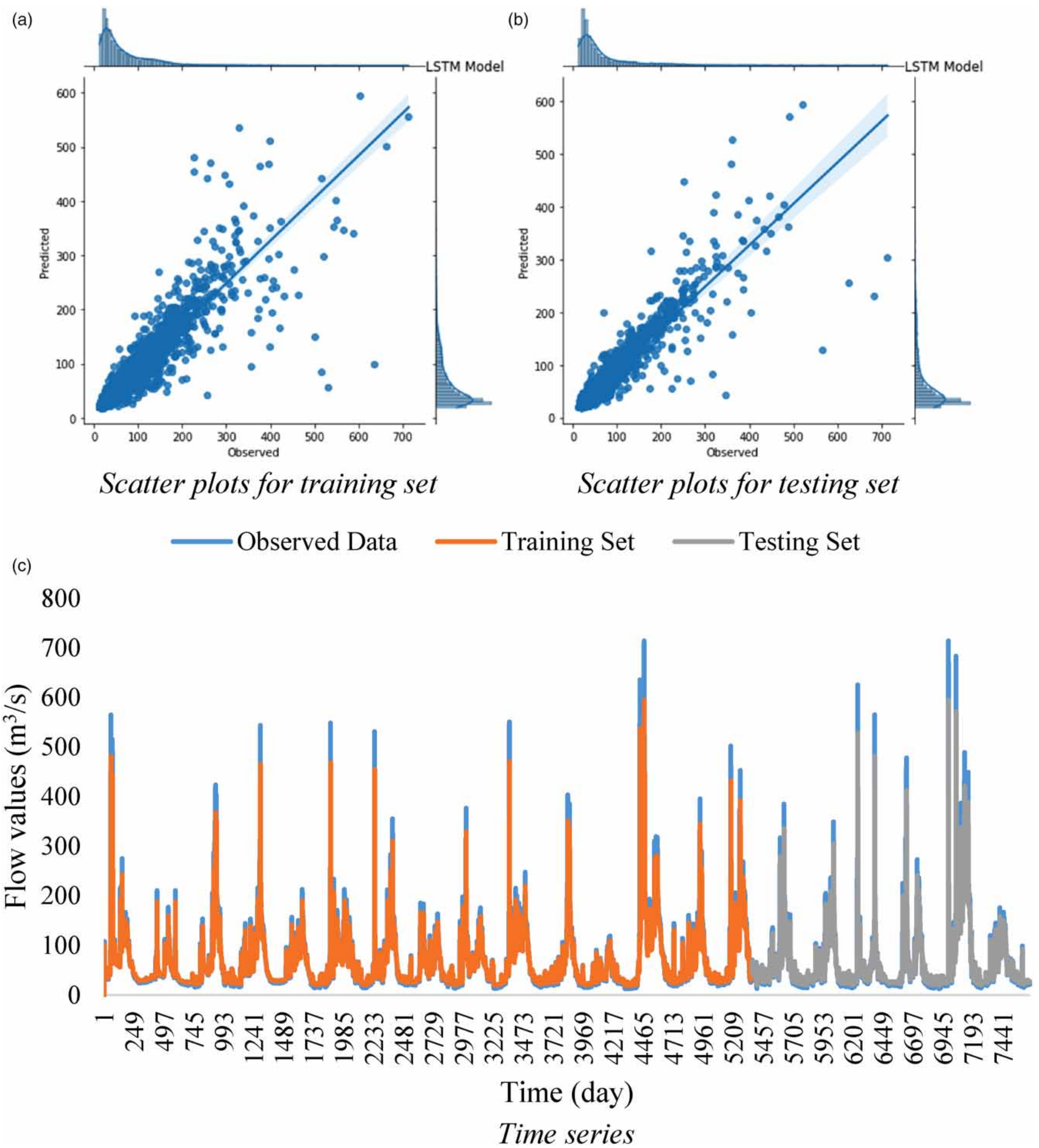


Figure 3 | LSTM model results.

Examining RFR models, model 5 has the best R^2 value (Table 3). R^2 values for training and testing sets of this model have been found as 0.918 and 0.871, respectively. All RFR models showed high performance except the model 6 as in the LSTM models.

When Table 4 was examined, model 5 showed better performance than other XGBoost models. RMSE values of the model having the lowest error values were obtained as 20.47 and 26.38 for training and testing sets, respectively. The R^2 value of the model is 0.871 for the testing set. In all methods, it was seen that the R^2 value of model 6 having inputs $Q_{(1719)t}$ and $Q_{(1720)t}$ increased by adding Q_{t-1} to these input parameters in model 7.

Upon evaluating the models, it is observed that the XGBoost model, utilizing five input parameters, demonstrates superior performance compared to the other models. The R^2 value of the XGBoost model for the testing set is determined to be 0.871, indicating its effectiveness in capturing the variance in the streamflow values. Scatter diagrams and time series of the best LSTM, RFR and XGBoost models are given in Figures 3–5 for training and testing sets, respectively.

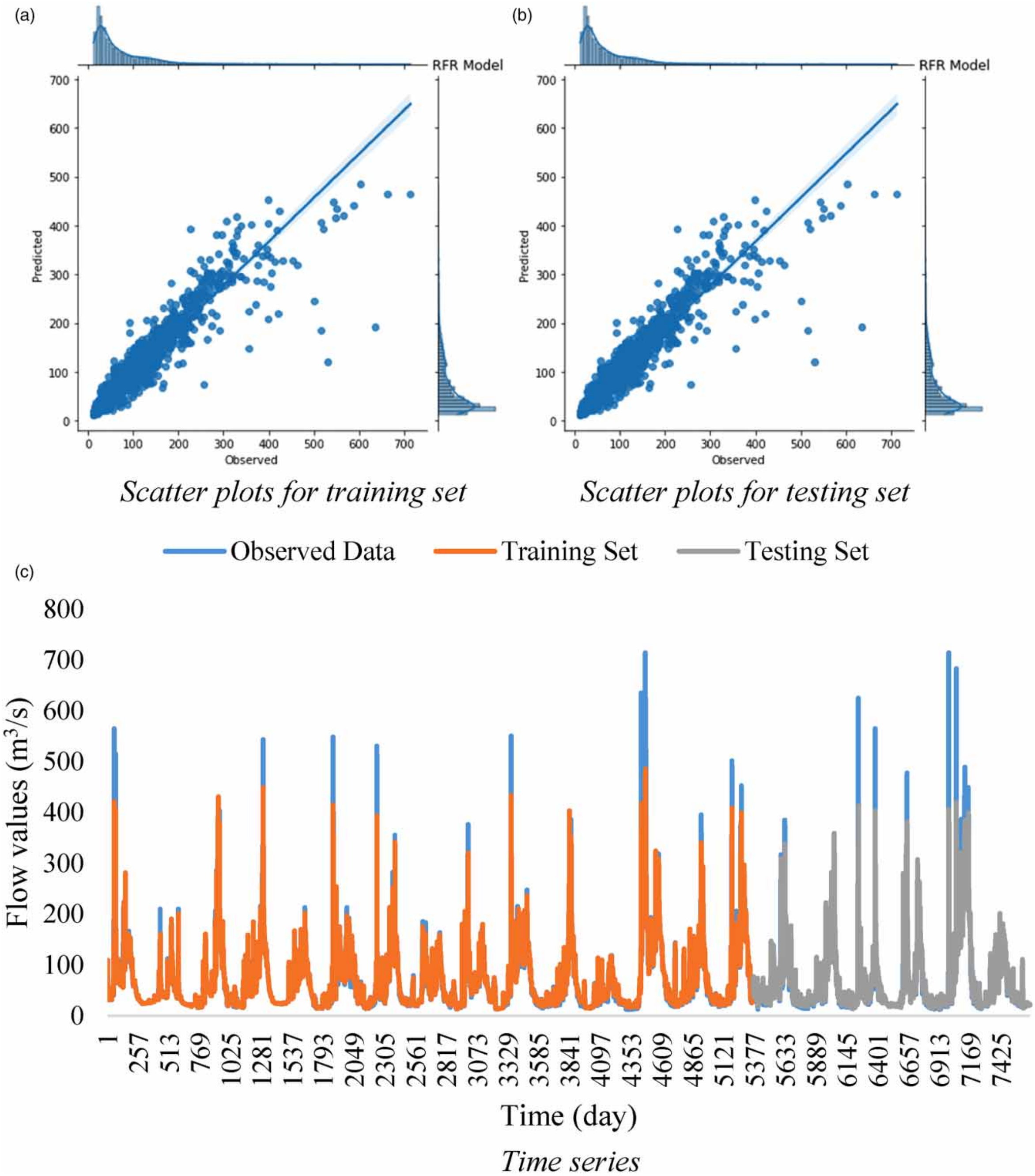


Figure 4 | RFR model results.

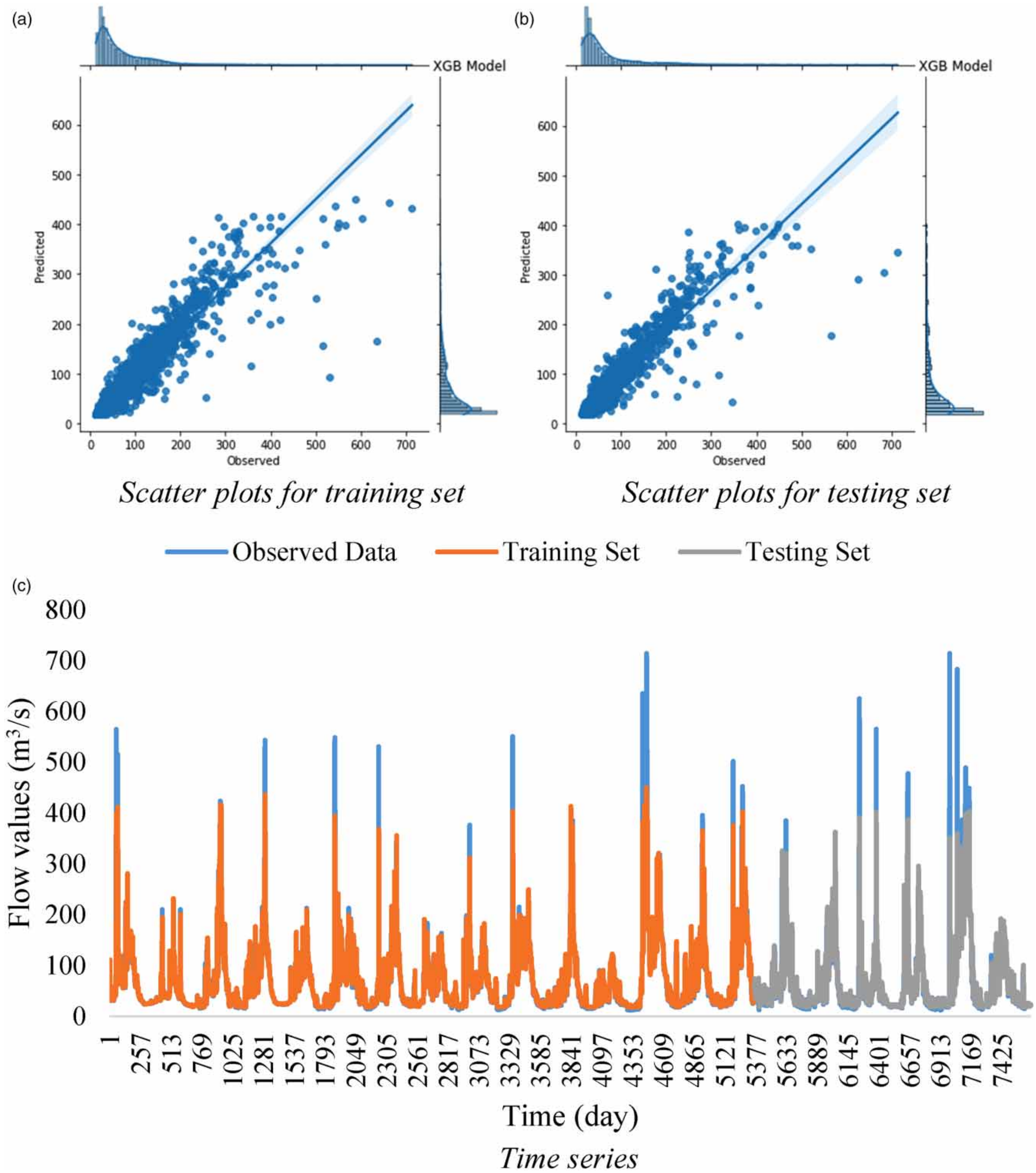


Figure 5 | XGBoost model results.

4. CONCLUSIONS

Accurate estimation of streamflow values is necessity for early disaster management in many parts of the world. However, it is very difficult to precisely predict the flow when streamflow measurements cannot be made. The slightest error in streamflow estimations can cause serious damage. Therefore, reliable flow estimation plays an important role in the planning of water resources. For these reasons, various flow estimation methods have been developed. The use of deep and machine learning in the development of estimation models has become inevitable, when it is thought that the number of data collected with the developing technology is increasing rapidly nowadays. The fact that studies on deep and machine learning continue with increasing momentum also supports this idea. For this reason, deep and machine learning was used in this study to estimate the flow of the Göksu River. Various LSTM, RFR and XGBoost models were developed by different input combinations. The hyperparameter tuning of the developed models was made and the best model structure was determined.

When the models are examined, it is seen that the XGBoost and RFR models perform close to each other, while the success of this model is more remarkable because the XGBoost model has lower error values for the testing set. The outcomes of this study underscore the successful application of deep and machine learning algorithms for accurate streamflow estimation. The XGBoost model showcases its potential in handling the complexities of the Göksu River's flow patterns. The findings contribute to the advancement of streamflow prediction methodologies, benefiting water resource management and disaster preparedness.

ACKNOWLEDGEMENTS

Onur Özcanoğlu, who was one of the potential authors of the manuscript, passed away during the preparation of this manuscript. The other authors are grateful to him for his great contribution to his paper.

DATA AVAILABILITY STATEMENT

Data cannot be made publicly available; readers should contact the corresponding author for details.

CONFLICT OF INTEREST

The authors declare there is no conflict.

REFERENCES

Alobaidi, M. H., Chebana, F. & Meguid, M. A. 2018 Robust ensemble learning framework for day-ahead forecasting of household based energy consumption. *Applied Energy* **212**, 997–1012. https://doi.org/10.1016/j.apenergy.2017.12.054.

Assem, H., Ghariba, S., Makrai, G., Johnston, P., Gill, L. & Pilla, F. 2017 Urban water flow and water level prediction based on deep learning. In: *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017*, September 18–22, Skopje, Macedonia. Proceedings, Part III 10. Springer International Publishing, pp. 317–329.

Aziz, K., Rai, S. & Rahman, A. 2015 Design flood estimation in ungauged catchments using genetic algorithm-based artificial neural network (GAANN) technique for Australia. *Natural Hazards* **77**, 805–821. doi:10.1007/s11069-015-1625-x.

Bai, Y., Chen, Z., Xie, J. & Li, C. 2016 Daily reservoir inflow forecasting using multiscale deep feature learning with hybrid models. *Journal of Hydrology* **532**, 193–206. https://doi.org/10.1016/j.jhydrol.2015.11.011.

Box, G. E. P. & Jenkins, G. M. 1970 *Time Series Analysis, Forecasting and Control*. Holden-Day, San Francisco.

Bender, M. & Simonovic, S. 1994 Time-series modeling for long-range stream-flow forecasting. *Journal of Water Resources Planning and Management* **120**(6), 857–870. https://doi.org/10.1061/(ASCE)0733-9496(1994)120:6(857).

Breiman, L. 1996 Bagging predictors. *Machine Learning* **24**(2), 123–140. https://doi.org/10.1007/BF00058655.

Budholiya, K., Shrivastava, S. K. & Sharma, V. 2022 An optimized XGBoost based diagnostic system for effective prediction of heart disease. *Journal of King Saud University – Computer and Information Sciences* **34**(7), 4514–4523. https://doi.org/10.1016/j.jksuci.2020.10.013.

Campolo, M., Soldati, A. & Andreussi, P. 2003 Artificial neural network approach to flood forecasting in the River Arno. *Hydrological Sciences – Journal–des Sciences Hydrologiques* **48**(3), 381–398. https://doi.org/10.1623/hysj.48.3.381.45286.

Chang, F. J., Chiang, Y. M. & Chang, L. C. 2007 Multi-step-ahead neural networks for flood forecasting. *Hydrological Sciences–Journal–des Sciences Hydrologiques* **52**(1), 114–130. https://doi.org/10.1623/hysj.52.1.114.

Chen, J., Jin, Q. & Chao, J. 2012 Design of deep belief networks for short-term prediction of drought index using data in the Huaihe River Basin. *Mathematical Problems in Engineering.* **2012**, 1–16. https://doi.org/10.1155/2012/235929.

Chen, T. & Guestrin, C. 2016 XGBoost: a scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, California, USA, pp. 785–794. https://doi.org/10.1145/2939672.2939785.

Cheng, M., Fang, F., Kinouchi, T., Navon, I. M. & Pain, C. C. 2020 Long lead-time daily and monthly streamflow forecasting using machine learning methods. *Journal of Hydrology* **590**, 125376. https://doi.org/10.1016/j.jhydrol.2020.125376.

Chiang, Y. M., Chang, F. J., Jou, B. J. D. & Lin, P. F. 2007 Dynamic ANN for precipitation estimation and forecasting from radar observations. *Journal of Hydrology* **334**(1–2), 250–261. https://doi.org/10.1016/j.jhydrol.2006.10.021.

Coulibaly, P., Anctil, F. & Bobée, B. 2000 Daily reservoir inflow forecasting using artificial neural networks with stopped training approach. *Journal of Hydrology* **230**(3–4), 244–257. https://doi.org/10.1016/S0022-1694(00)00214-6.

Dong, W., Huang, Y., Lehane, B. & Ma, G. 2020 XGBoost algorithm-based prediction of concrete electrical resistivity for structural health monitoring. *Automation in Construction* **114**, 103155. https://doi.org/10.1016/j.autcon.2020.103155.

Erdal, H. I. & Karakurt, O. 2013 Advancing monthly streamflow prediction accuracy of CART models using ensemble learning paradigms. *Journal of Hydrology* **477**, 119–128. https://doi.org/10.1016/j.jhydrol.2012.11.015.

Fan, J., Wu, L., Zhang, F., Cai, H., Zeng, W., Wang, X. & Zou, H. 2019 Empirical and machine learning models for predicting daily global solar radiation from sunshine duration: a review and case study in China. *Renewable and Sustainable Energy Reviews* **100**, 186–212. https://doi.org/10.1016/j.rser.2018.10.018.

Freund, Y. & Schapire, R. E. 1997 A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* **55**(1), 119–139. https://doi.org/10.1006/jcss.1997.1504.

Fu, M., Fan, T., Ding, Z. A., Salih, S. Q., Al-Ansari, N. & Yaseen, Z. M. 2020 Deep learning data-intelligence model based on adjusted forecasting window scale: application in daily streamflow simulation. *IEEE Access* **8**, 32632–32651. https://doi.org/10.1109/ACCESS.2020.2974406.

Galelli, S. & Castelletti, A. 2013 Assessing the predictive capability of randomized tree-based ensembles in streamflow modelling. *Hydrology and Earth System Sciences* **17**(7), 2669–2684. https://doi.org/10.5194/hess-17-2669-2013.

Girihagama, L., Naveed Khaliq, M., Lamontagne, P., Perdikaris, J., Roy, R., Sushama, L. & Elshorbagy, A. 2022 Streamflow modelling and forecasting for Canadian watersheds using LSTM networks with attention mechanism. *Neural Computing and Applications* **34**(22), 19995–20015. https://doi.org/10.1007/s00521-022-07523-8.

Hochreiter, S. & Schmidhuber, J. 1997 Long short-term memory. *Neural Computation* **9**(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735..

Hu, Y., Yan, L., Hang, T. & Feng, J. 2020 Stream-flow forecasting of small rivers based on LSTM. *arXiv preprint arXiv*:2001.05681. https://doi.org/10.48550/arXiv.2001.05681.

Jiang, H., He, Z., Ye, G. & Zhang, H. 2020 Network intrusion detection based on PSO-XGBoost model. *IEEE Access* **8**, 58392–58401. doi:10.1109/ACCESS.2020.2982418.

Jordan, M. I. 1986 Serial order: a parallel distributed processing approach. *Advances in Psychology* **121**, 471–495. https://doi.org/10.1016/S0166-4115(97)80111-2.

Jothiprakash, V. & Garg, V. 2009 Reservoir sedimentation estimation using artificial neural network. *Journal of Hydrologic Engineering* **14**(9), 1035–1040. https://doi.org/10.1061/(ASCE)HE.1943-5584.0000075.

Katipoğlu, O. M. & Sarıgöl, M. 2023 Coupling machine learning with signal process techniques and particle swarm optimization for forecasting flood routing calculations in the Eastern Black Sea Basin, Türkiye. *Environmental Science and Pollution Research* 1–18. https://doi.org/10.1007/s11356-023-25496-6.

Kentel, E. 2009 Estimation of river flow by artificial neural networks and identification of input vectors susceptible to producing unreliable flow estimates. *Journal of Hydrology* **375**(3–4), 481–488. https://doi.org/10.1016/j.jhydrol.2009.06.051.

Kostić, S., Stojković, M. & Prohaska, S. 2016 Hydrological flow rate estimation using artificial neural networks: model development and potential applications. *Applied Mathematics and Computation* **291**, 373–385. https://doi.org/10.1016/j.amc.2016.07.014.

Li, C., Bai, Y. & Chen, Z. 2016 Deep feature learning architectures for daily reservoir inflow forecasting. *Water Resources Management* **30**(14), 5145–5161. https://doi.org/10.1007/s11269-016-1474-8.

Li, Y., Liang, Z., Hu, Y., Li, B., Xu, B. & Wang, D. 2020 A multi-model integration method for monthly streamflow prediction: modified stacking ensemble strategy. *Journal of Hydroinformatics* **22**(2), 310–326. https://doi.org/10.2166/hydro.2019.066.

Ma, M., Zhao, G., He, B., Li, Q., Dong, H., Wang, S. & Wang, Z. 2021 XGBoost-based method for flash flood risk assessment. *Journal of Hydrology* **598**, 126382. https://doi.org/10.1016/j.jhydrol.2021.126382.

Mantovani, R. G., Horváth, T., Cerri, R., Vanschoren, J. & de Carvalho, A. C. 2016 Hyper-parameter tuning of a decision tree induction algorithm. In: *2016 5th Brazilian Conference on Intelligent Systems (BRACIS) Recife*, Brazil. IEEE, pp. 37–42. doi:10.1109/BRACIS.2016.018.

Modarres, R. 2007 Streamflow drought time series forecasting. *Stochastic Environmental Research and Risk Assessment* **21**(3), 223–233. https://doi.org/10.1007/s00477-006-0058-1.

Moghaddamnia, A., Gousheh, M. G., Piri, J., Amin, S. & Han, D. 2009 Evaporation estimation using artificial neural networks and adaptive neuro-fuzzy inference system techniques. *Advances in Water Resources* **32**, 88–97. https://doi.org/10.1016/j.advwatres.2008.10.005.

Mohammadi, B., Moazenzadeh, R., Christian, K. & Duan, Z. 2021 Improving streamflow simulation by combining hydrological process-driven and artificial intelligence-based models. *Environmental Science and Pollution Research* **28**, 65752–65768. https://doi.org/10.1007/s11356-021-15563-1.

Nguyen, T. T. 2015 An L1-regression random forests method for forecasting of Hoa Binh Reservoir's incoming flow. In: *7th International Conference on Knowledge and Systems Engineering (KSE)*, Ho Chi Minh City, Vietnam, pp. 360–364. doi:10.1109/KSE.2015.52.

Ni, L., Wang, D., Wu, J., Wang, Y., Tao, Y., Zhang, J. & Liu, J. 2020 Streamflow forecasting using extreme gradient boosting model coupled with Gaussian mixture model. *Journal of Hydrology* **586**, 124901. https://doi.org/10.1016/j.jhydrol.2020.124901.

Panagoulia, D. 2006 Artificial neural networks and high and low flows in various climate regimes. *Hydrological Sciences–Journal–des Sciences Hydrologiques* **51**(4), 563–587. https://doi.org/10.1623/hysj.51.4.563.

Partal, T. & Cigizoglu, H. K. 2008 Estimation and forecasting of daily suspended sediment data using wavelet–neural networks. *Journal of Hydrology* **358**, 317–331. https://doi.org/10.1016/j.jhydrol.2008.06.013.

Pham, B. T., Bui, D. T. & Prakash, I. 2017 Landslide susceptibility assessment using bagging ensemble based alternating decision trees, logistic regression and J48 decision trees methods: a comparative study. *Geotechnical and Geological Engineering* **35**(6), 2597–2611. https://doi.org/10.1007/s10706-017-0264-2.

Pramanik, P. & Panda, R. K. 2009 Application of neural network and adaptive neuro-fuzzy inference systems for river flow prediction. *Hydrological Sciences–Journal–des Sciences Hydrologiques* **54**(2), 247–260. https://doi.org/10.1623/hysj.54.2.247.

Rahimzad, M., Moghaddam Nia, A., Zolfonoon, H., Soltani, J., Danandeh Mehr, A. & Kwon, H. H. 2021 Performance comparison of an LSTM-based deep learning model versus conventional machine learning algorithms for streamflow forecasting. *Water Resources Management* **35**(12), 4167–4187. https://doi.org/10.1007/s11269-021-02937-w.

Rumelhart, D., Hinton, G. & Williams, R. 1986 Learning internal representations by error propagation, in parallel distributed processing. *Explorations in the Microstructure of Cognition* **1**, 318–362.
Salas, J. D. 1980 *Applied Modeling of Hydrologic Time Series*. Water Resources Publication, Colorado.
Schratz, P., Muenchow, J., Iturritxa, E., Richter, J. & Brenning, A. 2019 Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. *Ecological Modelling* **406**, 109–120. https://doi.org/10.1016/j.ecolmodel.2019.06.002.
Shahid, F., Zameer, A. & Muneeb, M. 2020 Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM. *Chaos, Solitons & Fractals* **140**, 110212. https://doi.org/10.1016/j.chaos.2020.110212.
Tao, Y., Gao, X., Hsu, K., Sorooshian, S. & Ihler, A. 2016 A deep neural network modeling framework to reduce bias in satellite precipitation products. *Journal of Hydrometeorology* **17**, 931–945. https://doi.org/10.1175/JHM-D-15-0075.1.
URL-1 2018a NumPy. Available from: http://www.numpy.org/ (accessed 18 July 2023).
URL-2 2018b Python data analysis library. Available from: https://pandas.pydata.org/ (accessed 18 July 2023).
URL-3 2018c Machine learning in Python. Available from: http://scikit-learn.org/stable/ (accessed 18 July 2023).
URL-4 2018d Keras: The Python deep learning library. Available from: https://keras.io/ (accessed 18 July 2023).
Uysal, G., Şorman, A. A. & Şensoy, A. 2016 Streamflow forecasting using different neural network models with satellite data for a snow dominated region in Turkey. *Procedia Engineering* **15**, 1185–1192. https://doi.org/10.1016/j.proeng.2016.07.526.
Valipour, M., Banihabib, M. E. & Behbahani, S. M. R. 2013 Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir. *Journal of Hydrology* **476**, 433–441. https://doi.org/10.1016/j.jhydrol.2012.11.017.
Van der Meer, D., Hoekstra, P. J., Van Donkelaar, M., Bralten, J., Oosterlaan, J., Heslenfeld, D., Faraone, S. V., Franke, B., Buitelaar, J. K. & Hartman, C. A. 2017 Predicting attention-deficit/hyperactivity disorder severity from psychosocial stress and stress-response genes: a random forest regression approach. *Translational Psychiatry* **7**(6), e1145–e1145. https://doi.org/10.1038/tp.2017.114.
Veintimilla-Reyesa, J., Cisneros, F. & Vanegas, P. 2016 Artificial neural networks applied to flow prediction: a use case for the Tomebamba River. *Procedia Engineering* **162**, 153–161. https://doi.org/10.1016/j.proeng.2016.11.031.
Venkatesan, E. & Mahindrakar, A. B. 2019 Forecasting floods using extreme gradient boosting – a new approach. *International Journal of Civil Engineering and Technology* **10**(2), 1336–1346.
Verma, A. & Mehta, S. 2017 A comparative study of ensemble learning methods for classification in bioinformatics. In: *7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*, Noida, Uttar Pradesh India, pp. 155–158. doi:10.1109/CONFLUENCE.2017.7943141.
Vogeti, R. K., Mishra, B. R. & Raju, K. S. 2022 Machine learning algorithms for streamflow forecasting of Lower Godavari Basin. *H2Open Journal* **5**(4), 670–685. https://doi.org/10.2166/h2oj.2022.240.
Wolpert, D. H. 1992 Stacked generalization. *Neural Networks* **5**(2), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1.
Wu, H., Cai, Y., Wu, Y., Zhong, R., Li, Q., Zheng, J., Lin, D. & Li, Y. 2017 Time series analysis of weekly influenza-like illness rate using a one-year period of factors in random forest regression. *Bioscience Trends* **11**(3), 292–296. https://doi.org/10.5582/bst.2017.01035.
Yaseen, Z. M., El-Shafie, A., Afan, H. A., Hameed, M., Mohtar, W. H. M. W. & Hussain, A. 2016 RBFNN versus FFNN for daily river flow forecasting at Johor River, Malaysia. *Neural Computing and Applications* **27**(6), 1533–1542. https://doi.org/10.1007/s00521-015-1952-6.
Yin, S., Tang, D., Jin, X., Chen, W. & Pu, N. 2016 A combined rotated general regression neural network method for river flow forecasting. *Hydrological Sciences–Journal–des Sciences Hydrologiques* **61**(4), 669–682. https://doi.org/10.1080/02626667.2014.944525.
Yu, X., Wang, Y., Wu, L., Chen, G., Wang, L. & Qin, H. 2020 Comparison of support vector regression and extreme gradient boosting for decomposition-based data-driven 10-day streamflow forecasting. *Journal of Hydrology* **582**, 124293. https://doi.org/10.1016/j.jhydrol.2019.124293.
Zemzami, M. & Benaabidate, L. 2016 Improvement of artificial neural networks to predict daily streamflow in a semi-arid area. *Hydrological Sciences–Journal–des Sciences Hydrologiques* **61**(10), 1801–1812. https://doi.org/10.1080/02626667.2015.1055271.
Zhang, M., Fulcher, J. & Scofield, R. A. 1997 Rainfall estimation using artificial neural network group. *Neurocomputing* **16**(2), 97–115. https://doi.org/10.1016/S0925-2312(96)00022-7.
Zhang, G. P., Patuwo, B. E. & Hu, M. Y. 2001 A simulation study of artificial neural networks for nonlinear time-series forecasting. *Computers & Operations Research* **28**(4), 381–396. https://doi.org/10.1016/S0305-0548(99)00123-9.
Zhang, X., Zhang, Q., Zhang, G., Nie, Z. & Gui, Z. 2018 A hybrid model for annual runoff time series forecasting using Elman neural network with ensemble empirical mode decomposition. *Water* **10**(4), 416. https://doi.org/10.3390/w10040416.
Zhao, X. H. & Chen, X. 2015 Auto regressive and ensemble empirical mode decomposition hybrid model for annual runoff forecasting. *Water Resources Management* **29**(8), 2913–2926. doi:10.1007/s11269-015-0977-z.
Zhu, S., Lu, H., Ptak, M., Dai, J. & Ji, Q. 2020 Lake water-level fluctuation forecasting using machine learning models: a systematic review. *Environmental Science and Pollution Research* **27**, 44807–44819. https://doi.org/10.1007/s11356-020-10917-7.

First received 19 July 2023; accepted in revised form 11 September 2023. Available online 22 September 2023