Testing statistical machine learning models with ARIMA for water level forecasting: The case of the Red River

Khoi Nguyen Ta, Van Hon Nguyen,Van Quoc Hoan Doan

Abstract

The ability to predict water levels is important in managing and preventing natural disasters as well as minimizing the effects of floods. Although physics-based models often provide good results, developing forecasting systems using them requires significant investment in computational effort and a wide range of hydrogeomorphic data. Furthermore, data-driven forecasting models are often simpler and faster to build. Statistical machine learning (ML) techniques have made significant strides in recent decades toward developing data-driven forecasting systems that deliver high performance and cost savings. Meanwhile, one of the famous linear statistical models for time series forecasting is autoregressive integrated moving average (ARIMA) and linear and nonlinear combined models. In this work, we examine and evaluate single methods as well as combined methods that benefit from both linear and nonlinear models. The proposed techniques for water level forecasting are ARIMA regression models, hybrid models and other machine learning statistical models. The effectiveness of different techniques was evaluated using hourly data collected from 2008 to April 2015 from the Hung Yen hydrological station and three-hour sampling of the Red River at Vu Quang, Hanoi from 2008 to 2017. The actual experimental results of these three huge data sets show the most specific and accurate assessment of the models we have chosen*.

Keywords: Water Level Forecasting, Hybrid Model, Statistical Machine learning, ARIMA, Long-term Univariate Time Series

*Model chosen by Combining Statistical Machine Learning Models with ARIMA for Water Level Forecasting: The Case of the Red River

March, 2024

1. Introduction

Forecasting river and lake levels accurately is crucial for managing water resources and issuing flood alerts. Researchers frequently employ time series hydrological prediction models to predict future data because water level data from hydrological stations frequently have a time series structure. It is possible to uncover hidden information by using historical data to forecast future water levels and behavior. This knowledge is crucial for managing water resources, lessening the effect of floods, and preventing or limiting natural catastrophes.

Water level forecasting models may be constructed using two methods, according to research: process-based and data-based numerical prediction. Since testing machine learning models to see which ones are the most accurate at predicting water level is our goal. Conversely, process-based models such as HEC-RAS (2017), MIKE-11 (2009, 2019), MIKE-21 (2013, 2018), and others frequently yield precise outcomes and comprehensively depict the characteristics of physical events. They need a variety of hydrogeomorphic data, including topography, geology, conductive roughness, and cross-sections, and are computationally costly. They also need massive amounts of data, which might be challenging to accommodate in the case of water levels and flows, and enough experience to evaluate the results. As a result, we have decided to use data-based methodologies to construct a forecasting model. Using just (most of) the information from the available data, data-driven prediction models seek to uncover correlations between characteristics or hidden information in the data. They are helpful for precise water level predictions and real-time river/lake flow forecasting since they are often quicker and simpler to build. The foundation of data-driven models is data, which is created via statistical and machine learning methods.

The time series of hydrological data frequently combine linear and non-linear components due to the effect of several external causes. Of all the linear statistical models for time series forecasting, autoregressive integrated moving average (ARIMA) is one of the most often used and effective (Ghimire, 2017; Birylo et al., 2018). Statistical machine learning (ML) techniques have been extensively utilized in the development of forecasting models for nonlinear time series (Wu et al., 2009; Peng et al., 2017; Yaseen and al., 2018; Gjika et al., 2019). Enhancing time series forecasting by combining these two methods makes logical; in fact, several studies in the literature on this combined strategy provide positive results. Pannakkong & Associates, 2017; Zhong et la., 2017a; Peng et al., 2017; Mohan and Reddy, 2018; Yaseen et al., 2018; Mousavi-Mirkalaei and Banihabib, 2019; Temr et al., 2019; Xie and Lou, 2019). We were motivated to adopt this approach to develop water level forecasting models and test them on the Red River scenario by the hybrid models' performance in several application areas. Specifically, we suggest a two-part data-driven hybrid modeling process that uses statistical learning non-parametric and ARIMA to model the linear and non-linear components of water level time series. We hypothesise that hidden patterns in time series data may be better exposed than with a single model approach by utilizing two types of models to capture linear and non-linear components in water level time series. By utilizing three sizable actual data sets from three hydrological stations to solve the Red River water level forecasting problem, we will test and experimentally validate our theory.

The rest of this paper is organized as follows: The next section overviews the related works in the literature. Sections 3 describes in details our proposed method. Data representation, pre-processing, evaluation metrics, and experiment results are pre- sented and discussed in Section 4. The last section (section 5) concludes the paper.

2. Related Works

One of the most popular and successful linear statistical models for time series forecasting is ARIMA. Three parameters—surface runoff, evapotranspiration, and precipitation—were used by the authors of (Birylo et al., 2018) to estimate ground water level using the ARIMA model. The twelve-month prediction results indicated that ARIMA models performed well. Similar to this, Ghimire (2017) created the ARIMA model, which proved to be highly successful in predicting US river flows. The work of Mirza-vand and Ghazavi (2015) demonstrated the effective use of five time series models to estimate groundwater level: moving average (MA), auto regressive moving average (ARMA), ARIMA, and seasonal ARIMA (SARIMA), as well as a combination of several time series models. Notably, the experiment findings showed that projections of ground water levels were significantly more accurate when time series models were combined. In order to anticipate the monthly inflow of the Dez dam reservoir, Valipour et al. (2013) compared the prediction ability of ARMA, ARIMA, and autoregressive artificial neural networks (ANN). The findings unequivocally demonstrated that during the last 12 months, the ARIMA model outperformed ARMA, while over the previous 60 months, autogressive ANN outperformed it. In order to forecast the daily water level of three stations in the middle reaches of the Yangtze River, Yu et al. (2017) looked into the ARIMA model. In order to anticipate the water level of the Mekong River, Nguyen et al. (2015) assessed the effectiveness of three statistical machine learning models: LASSO, Random Forest (RF), and Support Vector Regression (SVR). For a 5-lead-day, SVR yielded decent results with a mean absolute error of 0.486(m). An acceptable error range for a flood forecasting model is between 0.5 and 0.75 meters. In order to estimate the water level at two distinct stations in the Philippines' Cagayan River basin, Garcia et al. (2016) looked into the RF algorithm. The technique's favorable prediction performance was demonstrated by the correlations between the anticipated water level and ground truth data. These correlations also revealed that the approach may be applied to additional stations located on the major river basins in the Philippines. Pasupa and Jung-Jareantrat (2016) used a variety of statistical machine learning techniques, including linear regression (LR), kernel regression (KL), support vector regression (SVR), K-Nearest neighbor (KNN), and random forest (RF), to forecast the water level on the Chao Phyra river in Thailand. Better than the Royal Thai Navy's previous method, the SVR model with radial basis function kernel and 72-hour historical time series data produced the best prediction results with the least amount of error. SVR's capacity to forecast river flows was also shown in several studies (Garsole and Rajurkar, 2015; Adnan et al., 2018; Bafitile and Li, 2019). Yang et al. (2017) used five statistical machine learning techniques (RF, Kstar, RBF network, KNN, and Random Tree) to anticipate the water level time series on Taiwan's Shimen Reservoir. The experimental findings demonstrated that RF outperformed the other approaches in predicting performance. When it came to daily water level prediction, RF outperformed other statistical machine learning techniques including SVM, ANNs, and decision trees (DT) (Wang et al., 2018; Choi et al., 2019). Hipni et al. (2013) used Support Vector Machines

(SVM) in lieu of the Adaptive Neuro Fuzzy Inference System (ANFIS) to anticipate the daily dam water level of the Klang gate in Malaysia. The outcomes made it very evident that SVM outperforms ANFIS. In the other study, seasonal autoregressive model (SAR), multilayer perceptron (MLP), and support vector machines (SVM) were used to forecast long-term lake water level (Khan and Coulibaly, 2006). SVM is the most effective of these three methods. In the field of hydrology, ANNs have been widely used as a statistical machine learning approach for tasks including water flow modeling, water quality assessment, and water level forecasting (Toro et al., 2013; Kim and Seo, 2015; Kasiviswanathan et al., 2016; Hamid et al., 2019).

To increase modeling accuracy and flexibility, hybrid models can leverage the strengths of each component model (Zhang, 2003). To improve the hydrological time series forecast performance, several hybrid models have recently been proposed (Di et al., 2014; Seo et al., 2015; Zhong et al., 2017a; Yaseen et al., 2018; Chen et al., 2018; Yaseen et al., 2018; Rezaie-Balf et al., 2019; Mousavi- Mirkalaei and Banihabib, 2019; Nazir et al., 2019). In 2015, Seo and colleagues studied two hybrid models: the wavelet-based adaptive neuro-fuzzy inference system (WANFIS) and the wavelet-based artificial neural network (WANN). It has been demonstrated that WANN is a useful tool for forecasting water levels and produces superior outcomes than other traditional forecasting models. A hybrid ANN-Kalman filtering was suggested in (Zhong et al., 2017b) to forecast the daily water level for the MaAnshan station. When compared to ANN, the hybrid model's predicting outcomes were generally positive. The authors of (Mousavi-Mirkalaei and Banihabib, 2019) used nonlinear auto-regressive exogenous and ARIMA to estimate daily Urban Water Consumption (UWC) for Tehran Metropolis. The results of the studies unequivocally demonstrated that the hybrid model—which integrated both linear and nonlinear models—had a superior predicting accuracy for UWC. In order to anticipate water levels, Xu et al. (2019) developed a hybrid model that combines ARIMA and RNN (Recurrent Neural Network). The combined model is able to represent the general trend and amplitude variation more accurately, according to the experimental data. Another combination of SVR and ARIMA was suggested to forecast the Liuhe station's daily average water level data. The suggested approach demonstrated strong data anti-jamming performance and high predicting accuracy for water levels.

Motivated by the success of hybrid forecasting models, we propose a hybrid approach that combines ARIMA with different statistical machine learning methods to capture the linear and non-linear components of the time series separately. The novel method is then tested on the real datasets of the Red river for hourly water level forecasting.

3. Methods

Autoregressive Integrated Moving Average model, ARIMA

ARIMA, introduced in (Box and Jenkins, 1976), is one of the most popular statistical linear models for forecasting univariate time series. The main idea is that time series can be decomposed into present, past values and random errors. Hence, ARIMA is a combination of auto-regression AR(p) (an additive linear function of p past observations), moving average MA(q) (q random errors), and d which is an integer making a series to be stationary. The ARIMA(p, q, d) can be represented as:

$$\Delta^d y(t) = c + \sum_{j=1}^{p} \alpha_j \, \alpha_j \times y(t - j) + \epsilon(t) + \sum_{j=1}^{q} \beta_j \, \beta_j \times s(t - j) \qquad (1)$$

Where $\Delta = (1 - B)$, B is the 'Backward' operator and $By(t) = y(t - 1)$, $y(t)$ is the ob- servation data at time t, c is the constant, $\alpha_1, ..., \alpha_p$ are the auto-regressive parameters, $s(t)$ is the white noise at time t, and $\beta_1, ..., \beta_q$ are the moving average coefficients.

Fitting an ARIMA model involves 4 steps as follows:

- Identification of the ARIMA(p, d, q) structure;
- Estimation of parameters;
- Diagnostic checking on the estimated residuals;
- Forecasting future values based on the known data.

The autocorrelation function (ACF) and partial autocorrelation function (PACF) of the data are used to find the q and p orders of the ARIMA model (Box and Jenkins, 1976). The strength of ARIMA is the ability to turn non-stationary time series into stationary time series by differentiating d times. Stationarity is necessary because it makes prediction practical and useful. Therefore, before fitting the ARIMA model, we often need to transform the data if the data contains heterogeneity of the trend component. A time series is stationary when its mean, variance, and covariance (at different lags) remain constant over time. The coefficients α and β are estimated such that the overall error measure is minimal. In the diagnostic checking step, the statistical assumptions about the error of the model are checked using several diagnostic statistics and plots of the residuals.

Random Forest

Random forest (RF), proposed in (Breiman, 2001), is one of the most successful learning methods. It is an extension of bagging (bootstrap aggregation) to build different trees to develop the idea of random subspace sampling. Random forests consist of many individual trees, each built on a random sample of training data. The algorithm produces fully mature trees without pruning to keep bias low. Each random forest tree is learned on a set of bootstrap samples, and at each node, a subset of random attributes is considered for separation.

Randomness creates diversity among trees and allows control of low correlation between trees in the forest. So naturally, they are more accurate and stable as more trees are added.

The Random Forrest algorithm (Fig. 1) can be briefly described as follows:

- From a training dataset (LS), with m samples and n variables (features), construct independently T decision trees.
- The t^{th}-decision tree model is built on the t^{th}- bootstrap sample set from LS.
- At each inner node, randomly choose n^J variables ($n^J << n$) and calculate the best partition based on these n^J variables.
- Un-pruned trees are built with the maximum depth.

Then, each tree provides a prediction value y and the final prediction value is obtained by aggregating the results given by T individual trees from the forest. The RF prediction is

$$\hat{y} = \frac{1}{T} \sum_{i=1}^{T} \hat{f}_i(x) \tag{2}$$

where, $x \in R^n$ is a new input, T is the number of trees in the forest, $\hat{f}_i(x)$ is the prediction of unknown value y of input x generated from the i tree ($i = 1..T$).

When building a tree, an additional tuning parameter is the number of candidate variables selected for node splitting at each iteration, called *mtry*, and $1 < mtry < n$. An objective of selecting $mtry < n$ is to reduce the computational time. Normally, *mtry* is chosen as $mtry = \sqrt{n}$ for classification problems and $mtry = n/3$ for regression ones.

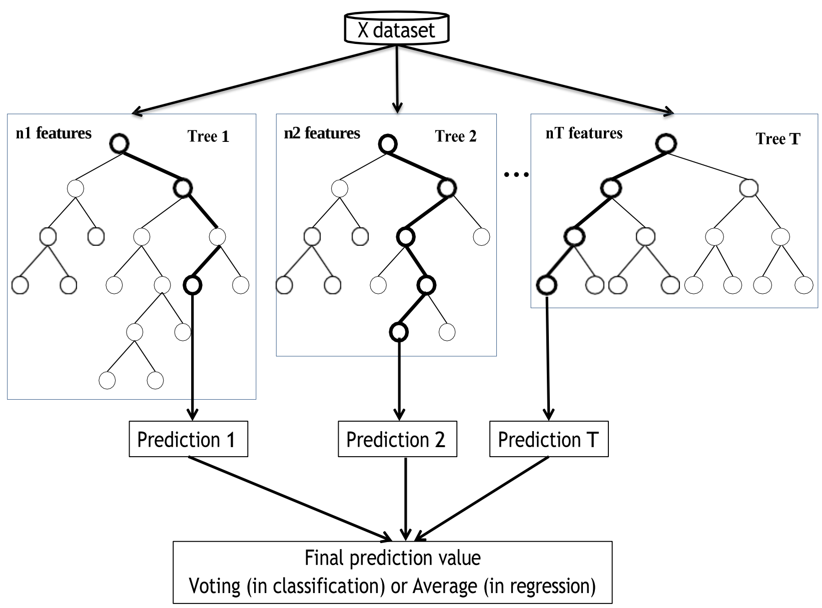


Figure 1: Diagram of the Random Forest building process.

Another parameter to consider is *ntree*. Trees usually tend to be unstable with high variance. In the random forest algorithm, number of trees (*ntree*) is used to reduce the variance.

Support Vector Regression

Support Vector Regression (SVR) is the regression version of Support Vector Machines (SVM), a statistical machine learning algorithm based on statistical learning theory developed in (Vapnik, 1995). The basic idea of SVM is to find an optimal hyper-plane for linearly separable patterns in a high dimensional space where features are mapped onto. There are more than one hyper-plane satisfying this criterion. The task is to discover the one that maximizes the margin around the separating hyper-plane (Fig. 2). This is done with the helps of the support vectors which are the data points that lie closest to the decision surface and have direct bearing on the optimum location of the decision surface.

The linear regression estimating function can be illustrated as follows:

$$f(x) = w^T x + b \tag{3}$$

where w is the weight vector, b is the bias and x is the input vector. SVMs can be extended for classification or regression problems that are not linearly separable by transforming original data into a new space using Kernel functions. The new space, called feature space, is usually high dimensional so that the classes become linearly separable.

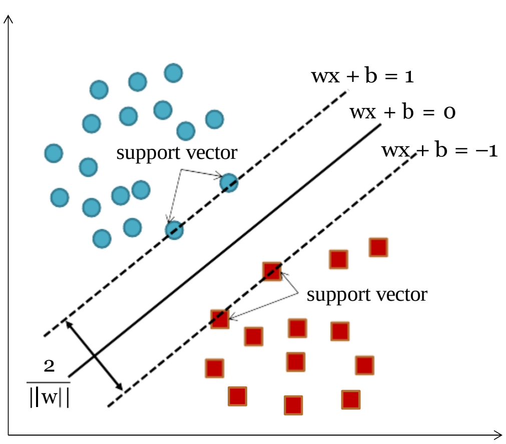


Figure 2: Linear classification with SVM

K-Nearest Neighbors

K-nearest neighbors (Altman, 1992), a (non-parametric) statistical learning method, has been widely used for both classification and regression problems. This algorithm predicts values of any new data points using a similarity measure (i.e distance function). This means that a new data sample is assigned a value based on how close it is to the points in the training set. In detail, for each sample of a test set, we compute the distance (e.g Euclidean, Manhattan, or Minkowski) between that sample and all samples in the training set and then we choose the *K* closest training data samples. The prediction value is the average of the target output values of these *K* nearest neighbors.

The proposed method

Time series data consists of linear and nonlinear components, with no universal model capable of modeling both. Linear models like ARIMA are sufficient for linear components, while non-parametric models like SVM, RF, and KNN can model nonlinear components. Hybrid models are proposed for better forecasting results, with arguments and justifications provided.

4. The results (not shown in full here to save the space) consistently confirmed the superiority of ARIMA based on the performance indicators described in section 4.2. For instance, Table 1 depicts the average forecast errors of ARIMA, PARMA, GAM- RMA for 12h-ahead on Hanoi time series. It is clear that, ARIMA was significantly better than the others on all performance indicators. Therefore, in this paper, ARIMA is chosen to develop our proposed hybrid approach.

To capture the non-linear component of time series, RF, SVM, KNN methods are utilized. We have selected these methods for a number of reasons. First, they are all (non-parametric) statistical learning methods that have relatively firm the- oretical foundations (eg. with theoretical guarantee on learnability, learning consis- tency, and universality). Second, they are representatives of large and popular classes of statistical machine learning techniques - lazy learning (KNN), kernel based methods (SVM), and ensemble learning methods (RF). Third, they have been successfully applied in modeling/ learning hydrological time se- ries (Nguyen et al., 2015; Yang et al., 2017; Wang et al., 2018; Choi et al., 2019; Hipni et al., 2013; Khan and Coulibaly, 2006; Fernandez-Delgado et al., 2014).

Details of the proposed method

Fig. 4 describes all steps of our proposed method for forecasting univariate time series. The objective of the proposed method is to take into account the advantages of the different forecasting models in terms of type (i.e. linear and non-linear) and complexity (i.e. single model and hybrid model).

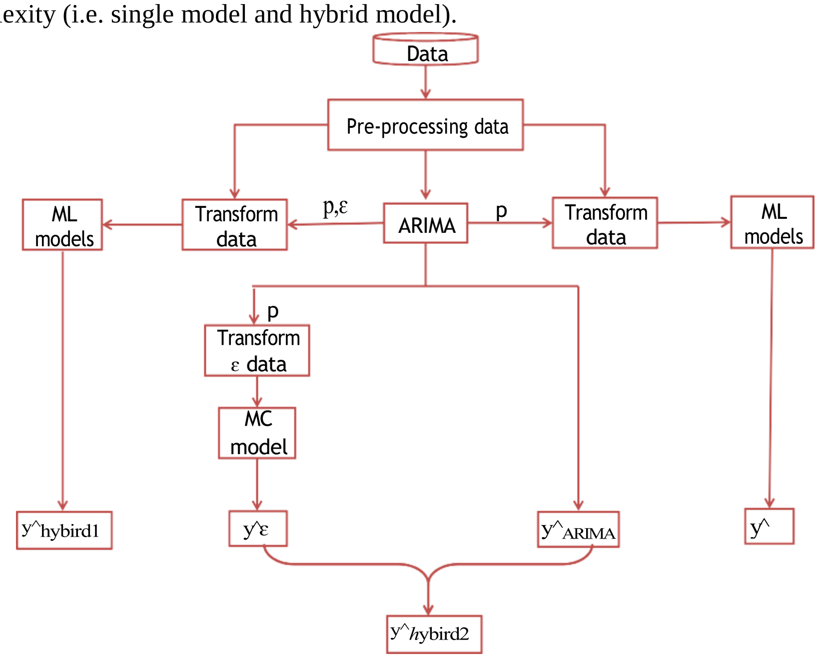


Figure 4: Flowchart of the proposed method for univariate time series forecasting

A time series can be considered as a combination of two components: linear and nonlinear ones (Zhang, 2003). That is,

$$y(t) = L(t) + N(t) \tag{4}$$

where $L(t)$ and $N(t)$ are the linear and nonlinear components of the time series. Both components (i.e. $L(t)$, $N(t)$) must be estimated from time series data.

The selection method implemented consists of three stages: i) linear model, (ii) linear-nonlinear model and (iii) forecast of future values. In the first stage, we apply the ARIMA model to extract the linear part of the time series. Then, in the second stage, the residuals obtained from ARIMA and the lag values of the time series are used as input to train statistical machine learning models. Finally, in the third stage, future values are estimated from different hybrid models. Details of these steps are described as follows:

- Linear model ARIMA is conducted on the entire time series to obtain the predicted values Lˆ(t) and their residuals. The residual of this step is used as an input in the second stage (hybrid model). Specifically, in this study, we use the p-order obtained from the ARIMA model to determine the optimal delay time for the input of the hybrid models. This allows converting one-dimensional time series into p-dimensional data so that statistical machine learning methods can be applied to univariate time series forecasting.
- Linear-nonlinear model Residuals are important in selecting appropriate linear models because a model is not completely linear if it still contains nonlinear components in its residuals. Usually, we cannot detect any nonlinear components when analyzing residuals, and in fact, there is no statistical method that allows us to identify nonlinear autoregressive relationships (Mousavi -Mirkalaei and Banihabib, 2019). Therefore, we find that accounting for residuals in hybrid models is accomplished by using statistical machine learning models to explore nonlinear relationships in the residuals.

The residual from ARIMA model is calculated by:

$$\epsilon(t) = y(t) - \hat{L}(t) \tag{5}$$

where s(t) is the residual and $\hat{L}(t)$ is the forecast value of ARIMA model at time t. In order to discover the nonlinear relationships of the time series, s(t) is could be fitted by statistical machine learning models such as KNN, SVM, RF. In this study, we build and test two types of hybrid models as follows:

– Model 1

$$y\hat{}_{hybrid1}(t) = f(y(t-1), ..., y(t-p), \epsilon(t-1), \epsilon(t-2)) \tag{6}$$

where f a nonlinear function learnt from data by the machine learning mod- els. Here, we perform some pre-experiments on our data to find an appro- priate stop of s with different values (1, 2, 3, ..., p). The results showed that when the past window is greater than 2 the predicted values are not better and it requires longer computational time. Therefore, the past window of s is empirically chosen as 2 in this paper.

- Forecasting: The models built in the previous step are used to forecast water levels (in our study, for the Red River). Each model has characteristics and applications suitable for each type of data and specific problem. Choosing the appropriate model requires an understanding of the nature of the data and the goal of the forecasting problem.

Next we come to building and calculating parameters for evaluating models and then going into the actual evaluation process on each data set to make the best choice.

4. Experimental Results and Discussions

In this section, we first describe the datasets and their pre-processing stage in our experiments, then, define the performance evaluation metrics, and finally discuss the experimental results with key findings and observations.

4.1. Data representation and pre-processing

In this study, water level data from three hydrological stations of the Red River (the main river in the Northern Delta of Vietnam), Vu Quang, Hanoi and Hung Yen, are used to test machine learning methods. hybrid method as we have selected and analyzed above. We will conduct test runs on three data sets to derive the most optimal parameters for each model group and compare the accuracy with the article Combining Statistical Machine Learning Models with ARIMA for Water Level Forecasting: The Case of the Red River (Phan et al., 2020). After testing the model coefficients, we will go to the next step which is to provide overall results of the models and continue to find the model. best for the problem. In Figure 5, we proceed to analyze the time series elements, proceeding to continue with the parts related to them in separating the data into training and testing sets.

Vu Quang, Hanoi and Hung Yen. Here are some general comments about the chart:

Vu Quang: It seems that this site experiences greater fluctuations in water level than the other two sites, with clear peaks each year. This could indicate the annual rainy season and flooding, or geographical features could be responsible for large fluctuations in water levels.

Hanoi: The chart shows that the water level also fluctuates quite significantly, but not as high as at Vu Quang. A steady increase in water levels can be seen every year, especially in certain years, such as around mid-2010 and late 2016, indicating possible heavy rain events or flooding.

Hung Yen: The water level here is less volatile than the other two locations. However, there are still sudden increases, although not as high as Vu Quang or Hanoi. This may indicate that Hung Yen is not affected by the rainy season as much as the other two places or has a more effective water management system.

- Vu Quang and Hanoi datasets:
  The water level of these two datasets were collected at the Vu Quang and Hanoi hydrology stations in Vietnam. The samples were taken from 01 January 2008 to 31 December 2017 with different frequencies. The reason is that on normal days without rainfall or flood, the frequency of sampling is low, for example, with 4, 6, 7 or 8 times per day or no data available. However, on rainy days, the frequency of sampling is much more such as with 10, 11, 12, 18, 19, 20, 21, 22, or even 23 times a day. Therefore, before applying time series forecasting algorithms to predict the water level, it is necessary to re-sample data at equal time intervals and pre-process them. For these time series, We normalized the sampling frequency to 8 times per day at 1h, 4h, 10h, 13h, 16h, 19h, 22h. Consequently, there are two cases:

  - *The data sampling a day is less than 8 times or no data.*
    In this case, we consider the periods of non-sampling or days without data as missing data. Then, we use imputation methods such as linear interpolation and/or moving average to fill in the missing data. For the large consecutive missing data (i.e. a full day missing or more), the result will be a straight line when applying linear interpolation. So in order to capture the dynamism of the data, we utilize DTWBI method (Phan et al., 2017) that allows to complete consecutive missing data while taking into account the dynamism of data.
  - *The data sampling a day is more than 8 times.*
    We base on the sampling times to recalculate the water level with the time aforementioned. This means we perform re-sampling 8 times a day at 1h, 4h, 7h, 10h, 13h, 16h, 19h, and 22h.

  In addition to considering the number of samples collected, we also replace rows containing date information because some rows are missing this information in the data, and in addition, test repeating values to proceed with deletion, combining time values to easily apply models.
- Hung Yen dataset:

This data set was collected at Hung Yen hydrological station every hour from January 1, 2008 to April 23, 2015 (Figure 5). Every day with 24 data samples collected continuously.
First, we check for duplicate data and remove them. Pairing time values makes it easier to apply models.
It also contains some missing data points and we use interpolation to complete the missing points, to ensure periodicity, seasonality and temporal continuity in the data. There will be some unknown data points we proceed to replace it with the value 0.

The water levels at Vu Quang, Hanoi, and Hung Yen are characterized by significant fluctuations, with clear peaks each year. These fluctuations may be due to the annual rainy season, flooding, or geographical features. Hanoi experiences a steady increase in water levels, especially during heavy rain events or flooding. Hung Yen is less volatile but still experiences sudden increases, possibly due to a more effective water management system. Peak levels occur consistently across locations, reflecting common environmental factors and the cyclical nature of the rainy season.

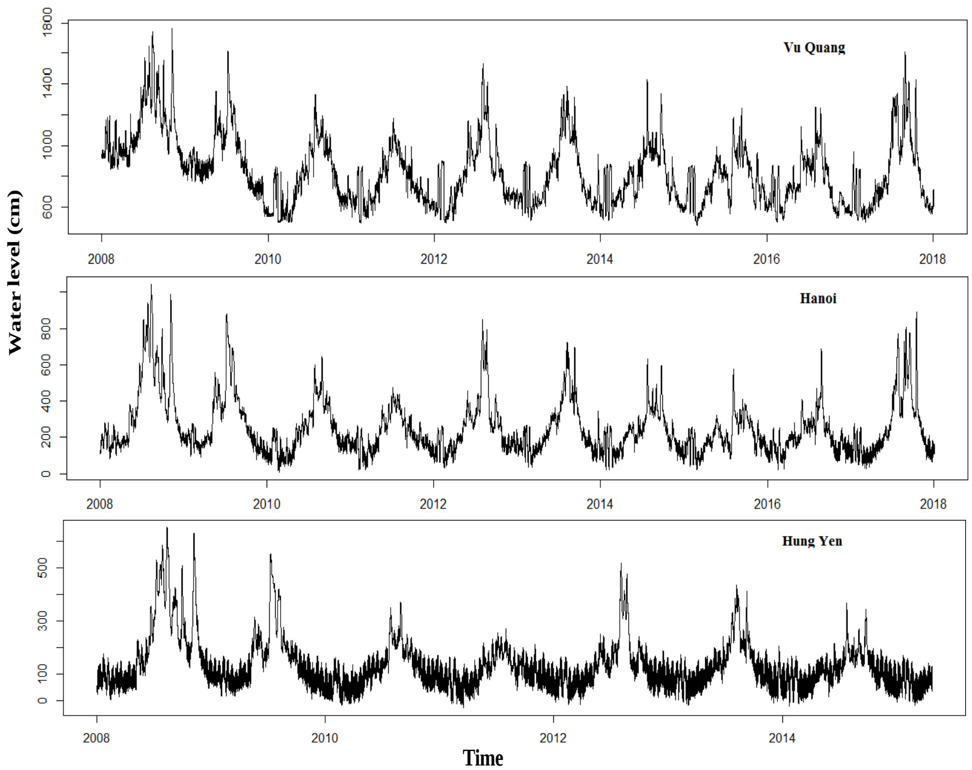


Figure 5: Water level at three hydrology stations of the Red river

The characteristics of the three datasets are summarized in Table 2.

No	Dataset name	Period	# Samples	Trend (Y/N)	Seasonal (Y/N)	Frequency
1	Vu Quang	2008-2017	29224	N	Y	3 hours
2	Hanoi	2008-2017	29224	N	Y	3 hours
3	Hung Yen	2008-2015	64061	N	Y	hourly

Table 2: Characteristics of the water level time series

4.2. Performance evaluation indicators

After the prediction phase, six evaluation metrics were used to assess different models, they are Sim, MAE, RMSE, *R* score, FSD and NSE. These metrics were selected as they possess different properties that are important to efficiently understand the performance of forecasting models from different angles. They are defined as follows:

1. Similarity: defines the percentage of similar values between the predicted values y and the actual values x. It is calculated as:

$$\text{Sim(x,y)} = \frac{1}{T}\sum_{i=1}^{T}\frac{1}{1+\frac{|y_i - x_i|}{\max(x)-\min(x)}} \tag{9}$$

Where T is the number of forecasting values. A higher similarity (Sim value $\in$ [0, 1]) highlights a better ability of the method for the forecasting task.

2. MAE: The Mean Absolute Error between the predicted values y and the actual ones, x, is computed as:

$$\text{MAE(y, x)} = \frac{1}{T}\sum_{i=1}^{T}|y_i - x_i| \tag{10}$$

A lower MAE value means better performance method for the prediction task.

3. RMSE: The Root Mean Square Error is defined as the average squared difference between the forecast values y and the respective true values x. This indicator is very useful for measuring overall precision or accuracy. In general, the most effective method would have the lowest RMSE.

$$\text{RMSE(y, x)} = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(y_i - x_i)^2} \tag{11}$$

4. R score: is determined as the correlation coefficient between two variables y and x. This indicator makes it possible to assess the quality of a forecasting model. A method presents better performance when its R score is higher ($R \in$ [0, 1])

5. FSD : The Fraction of Standard Deviation is defiend as

$$\text{FSD(y, x)} = 2 * \frac{|SD_{(y)} - SD_{(x)}|}{SD_{(y)} + SD_{(y)}} \tag{12}$$

5. NSE: The Nash Sutcliffe efficiency is used to evaluate the predictive ability of hydrological models. The NSE values range from -∞ to 1, with higher values mean better fit between observed and forecast water level (Nash and Sutcliffe, 1970).

$$\text{NSE} = 1 - \frac{\sum_{i=1}^{T}(x_i - y_i)^2}{\sum_{i=1}^{T}(x_i - \overline{x_i})^2} \tag{13}$$

4.3. Results and Discussions

To evaluate and compare all tested methods, the whole collected data were divided into two parts for training and testing. With Vu Quang and Hanoi time series, the training datasets were those from 2008 to 2015 accounted for 70% of the data, and the remaining observed samples (30%) from 2016 to 2017 were used to assess the forecasting models. With Hung Yen dataset, the data from 2008 to 2013 (about 70%) were used

for training models and the remaining data from 2014 to 2015 (about 30%) were utilized to test the learnt models.
When dividing a data set for a time series problem, the use of "timestep" is important to determine how information from the past is used to predict the future. The dataset is divided into two parts: training data and testing data.

Training data is used to build the model and testing data is used to evaluate the model. In predicting Red River water level, models such as KNN, SVR, RF and hybrid models have been tested. It is important that the model is trained not just on a previous data point, but on p previous values to predict the value at time t. When forecasting continues for the next time, the previous forecast value y(t) will be used as part of the input data for the next forecast y(t+1). With the ARIMA model, all historical values up to time t-1 are used to build the model. This means the model needs to be retrained every time new data becomes available to forecast at different times.

And to determine the time-step for the data set, we used a combination of methods such as determining data seasonality, delay between data collection and FFT method and went through many tests. Try to get the most optimal timestep for running and testing the model.

When using timesteps in dividing the data set, we must ensure that the model can learn from data samples over many time periods to be able to generalize and predict accurately for unknown data. seen in the future. Each "timestep" is essentially a step in the past that is used to make predictions. This technique helps the model learn to use information from the past to improve its accuracy in predicting the future.

This process is illustrated in Figure 6, where training data is used to build an ARIMA model at different times.

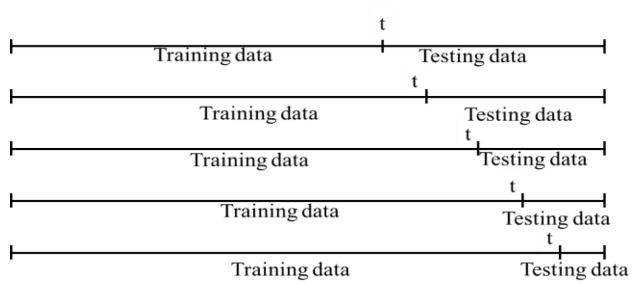


Figure 6: Training ARIMA model at different times

After running and testing the model, we have extracted the most optimal parameters for each model in addition to having a suitable timestep, respectively 168 (Hung Yen data set), 121 (Hung Yen data set), 121 (Hung Yen data set). Hanoi data set),390(Vu Quang data set). Table 3 below is a summary of the parameter values we obtained and it shows the error in the set p,q,d (2,1,5) in the ARIMA model for the Hanoi data set when compared with article* that we are currently researching.

Dataset name		Method	
	ARIMA (p,d,q)	KNN	RF
Vu Quang	(5,1,1)	K=6	mtry=4, ntree=100
Hanoi	(5,1,5)	K=7	mtry=4,ntree=1000
Hung Yen	(5,1,3)	K=5	mtry=4, ntree=300
		ARIMA_KNN	ARIMA_RF
Vu Quang		K=6	mtry=6, ntree=200
Hanoi		K=5	mtry=7,ntree=1000
Hung Yen		K=6	mtry=6, ntree=100
		SVM	
Vu Quang		C=1, gamma= 0.001, kernel= rbf	
Hanoi		C=50, gamma= 0.001, kernel=rbf	
Hung Yen		C=50, gamma= 0.01, kernel=rbf	
		ARIMA_SVM	
Vu Quang		C=50, gamma= 0.001, kernel=rbf	
Hanoi		C=5, gamma=0.001, kernel=rbf	
Hung Yen		C=50, gamma= 0.001, kernel=rbf	

Table3:Selected parameter sofvarious forecasting methods

Tables 4 and 5 provide detailed information on the average prediction results of all tested methods at 20 different random times for the Vu Quang, Hanoi and Hung Yen datasets. Different forecast ranges are used to evaluate the forecasting performance of the proposed hybrid methods and other methods. For the Vu Quang and Hanoi time series, forecast models are set up for 12h, 24h, 48h, 72h and 5-day periods. For Hung Yen data, these models are applied to predict the next 6h, 12h, 24h, 48h, 72h and 5 days. The best results for each forecast period are highlighted in bold. With the two data sets Hanoi and Vu Quang, it can be seen that hybrid methods using Model 1 (Formula 6) such as ARIMA-RF and ARIMA-KNN often give no better results when compared to other models. (i.e. ARIMA, RF, KNN, SVM), only a few benchmarks obtain superior results from hybrid methods.

From table 5, it can be seen that with the Vu Quang data set, the RF model achieves the lowest MAE, RMSE and highest Sim measurements in most prediction points (reaching the highest prediction peak within the prediction time). guess 48h).

For the Hanoi data set, we can see that the KNN model obtains the most optimal measurements. From there we can see the optimization of these methods. However, from the article* that we monitor and analyze, the hybrid methods have the most optimal measurements and are the best choices for the data. We may have encountered some errors in the process of slicing and dividing the data set When implementing hybrid methods, this confusion occurred.

However, with the Hung Yen data set, we observed uniformity from the methods and even discovered some points. Hybrid methods give much better performance than single methods. From there, we can draw some initial observations that in the process of dividing the residual data sets, we encountered some small errors, but the special thing here is that with the Hung Yen data set, The parameters seem to be perfect. We will continue to conduct testing, rebuilding the data residual division steps to obtain results closer to the research article*.

Table 4: Average results of various forecasting algorithms (hybrid approaches based on function 6 - Hybrid Model 1) on Hung Yen time series

Method	Sim	MAE	RMSE	FSD	R	NSE	Sim	MAE	RMSE	FSD	R	NSE
	6h						48h					
ARIMA	0.8	8.78	9.29	1.56	**0.99**	0.36	0.81	30.94	39.53	1.36	0.75	-0.34
RF	0.7	15.26	19.6	1.67	0.9	-1.04	0.88	17.42	20.48	**0.86**	0.96	0.64
ARIMA_RF	**0.81**	**8.24**	9.16	**1.59**	0.98	**0.38**	0.82	30.8	39.31	1.36	0.75	-0.33
KNN	0.65	19.83	21.7	1.92	0.68	-2.5	**0.9**	**14.86**	19.3	1.99	0.92	0.68
ARIMA_KNN	**0.81**	8.51	9.25	**1.59**	**0.99**	0.36	0.82	30.85	39.43	1.36	0.75	-0.34
SVM	0.74	12.24	13.02	1.95	0.96	-0.25	0.88	16.9	**18.07**	1.97	**0.97**	**0.72**
ARIMA_SVM	**0.81**	**8.24**	**9.11**	1.6	**0.99**	**0.38**	0.8	30.89	39.23	1.36	0.76	-0.32
	12h						72h					
ARIMA	**0.87**	13.44	15.00	**1.08**	**0.99**	0.7	0.8	33.49	41.55	1.46	0.68	-0.73
RF	0.8	22.09	23.71	1.15	**0.99**	0.25	0.89	15.93	18.92	**0.91**	0.95	0.64
ARIMA_RF	**0.87**	**13.06**	**14.78**	1.09	**0.99**	**0.71**	0.8	33.35	41.35	1.46	0.69	-0.72
KNN	0.77	26.51	28.13	1.98	0.97	-0.06	**0.9**	**13.38**	**17.17**	1.99	0.92	**0.7**
ARIMA_KNN	**0.87**	13.25	14.87	1.1	**0.99**	0.7	0.8	33.39	41.42	1.46	0.69	-0.72
SVM	0.85	15.1	15.69	1.98	0.98	0.67	0.87	19.4	20.68	1.99	**0.96**	0.57
ARIMA_SVM	**0.87**	13.18	14.91	1.1	**0.99**	0.7	0.8	33.39	41.26	1.46	0.7	-0.7
	24h						5 days					
ARIMA	0.82	27.07	33.29	1.18	0.9	0.07	0.77	41.16	47.82	1.57	0.6	-1.72
RF	0.86	19.16	22.87	**0.81**	0.93	0.56	0.88	18.45	23.75	**1.03**	0.85	0.33
ARIMA_RF	0.82	26.81	33.06	1.18	0.9	0.08	0.77	41.03	47.64	1.56	0.60	-1.69
KNN	0.85	20.94	24.99	1.99	0.91	0.47	**0.92**	**11.48**	**14.76**	1.99	**0.92**	**0.74**
ARIMA_KNN	0.82	26.96	33.22	1.18	0.9	0.07	0.77	41.03	47.68	1.56	0.60	-1.7
SVM	**0.9**	**13.55**	**15.08**	1.99	**0.97**	**0.81**	0.82	28.4	33.24	1.98	0.79	-0.31
ARIMA_SVM	0.82	26.92	32.97	1.17	0.91	0.09	0.77	41.09	47.62	1.56	0.61	-1.69

Table 5: Average results of various forecasting algorithms (hybrid approach based on function 6) on Vu Quang and Hanoi datasets

Method	Size	Vu Quang - Model 1						Hanoi - Model 1					
		Sim	MAE	RMSE	FSD	R	NSE	Sim	MAE	RMSE	FSD	R	NSE
ARIMA	12h	0.67	**4.39**	**5.41**	1.95	-0.036	-6.29	0.7	30.74	35.9	1.73	0.15	-3.24
RF		0.33	15.89	16.55	1.91	0.2	-67.15	**0.85**	11.7	14.38	1.96	0.71	0.34
ARIMA_RF		**0.78**	6.0	7.01	**0.64**	**0.93**	**-0.2**	0.81	**7.0**	**8.26**	**0.37**	**0.84**	**-0.10**
KNN		0.52	6.8	7.08	2.0	0.52	-11.48	0.71	28.86	33.67	1.8	0.17	-2.73
ARIMA_KNN		0.71	7.1	7.96	0.74	0.91	-1.5	0.71	28.72	32.57	1.79	0.48	-2.49
SVM		0.31	16.65	17.15	1.87	0.43	-72.14	0.73	12.38	29.52	1.93	0.44	-1.86
ARIMA_SVM		0.61	20.6	25.50	0.84	**0.93**	-43	0.69	31.39	35.94	1.8	0.19	-3.25
ARIMA	24h	0.79	15.92	22.47	1.95	-0.58	-1.49	0.72	46.41	52.56	1.75	0.28	-3.2
RF		0.78	14.81	15.86	1.91	0.43	-0.24	0.86	19.19	23.59	**1.67**	**0.82**	**0.15**
ARIMA RF		0.77	10.9	13.27	0.66	0.73	-1.1	0.72	45.32	51.42	1.8	0.3	-3.02
KNN		**0.84**	10.33	14.52	2.0	0.68	**-0.04**	**0.88**	**15.86**	**19.6**	1.97	0.78	0.41
ARIMA_KNN		0.76	**10.5**	**12.30**	**0.61**	**0.87**	-1.2	0.72	45.35	51.12	1.81	0.43	-2.97
SVM		0.76	15.86	16.74	1.92	0.77	-0.38	0.78	31.67	35.43	1.96	0.79	-0.9
ARIMA_SVM		0.59	57.6	76.60	0.93	0.67	-90	0.71	47.06	52.8	1.8	0.4	-3.24
ARIMA	48h	0.71	24.45	29.14	1.95	-0.61	-3.01	0.69	57.2	63.25	1.79	0.33	-3.93
RF		**0.81**	**13.23**	**15.04**	1.85	0.51	**-0.09**	**0.86**	20.42	25.87	**1.53**	0.71	**0.17**
ARIMA RF		0.80	18.1	22.21	0.70	0.74	-0.8	0.69	50.03	61.94	1.81	0.41	-3.73
KNN		**0.81**	13.79	17.54	1.07	-0.037	-0.49	0.89	**15.61**	**18.98**	1.98	**0.79**	0.56
ARIMA KNN		**0.81**	18.0	22.03	**0.69**	**0.88**	-0.6	0.7	55.24	61.12	1.79	0.43	-3.6
SVM		0.79	15.28	17.11	1.93	0.67	-0.4	0.82	28.43	33.81	1.97	0.72	-0.41
ARIMA_SVM		0.55	115.2	144.50	0.97	0.59	-161	0.69	57.49	63.26	1.79	0.41	-3.93
ARIMA	72h	0.69	26.44	30.16	1.96	-0.56	-4.11	0.69	57.78	63.27	1.82	0.38	-4.29
RF		0.81	**13.45**	**15.22**	1.85	0.52	**-0.3**	0.87	19.16	24.49	**1.53**	**0.67**	**0.2**
ARIMA RF		**0.82**	25.5	31.47	**0.72**	0.78	-0.8	0.69	57.05	62.34	1.79	0.43	-4.14
KNN		0.78	16.94	22.06	1.9	-0.41	-1.74	**0.88**	**16.3**	**20.34**	1.97	0.74	0.45
ARIMA KNN		**0.82**	25.7	32.41	0.78	**0.86**	-1.3	0.69	56.58	62.04	1.71	0.38	-4.09
SVM		0.73	21.31	24.12	1.96	0.52	-2.27	0.85	23.42	29.3	1.98	0.66	-0.13
ARIMA SVM		0.54	147.8	176.32	0.98	0.58	-105	0.68	58.02	63.31	1.77	0.4	-4.3
ARIMA	5 days	0.75	62.5	80.03	1.97	-0.37	**0.37**	0.67	72.4	78.4	1.86	0.3	-5.32
RF		0.83	36.31	54.45	1.84	0.44	-0.09	0.81	36.53	46.61	**1.57**	0.16	-1.23
ARIMA RF		0.80	37.41	44.38	**0.77**	0.79	-1.0	0.67	70.07	78.18	1.8	0.23	-5.28
KNN		0.78	57.61	89.96	1.99	**0.83**	-2.32	0.83	31.35	40.36	1.97	0.34	-0.67
ARIMA KNN		0.81	37.43	45.59	0.82	0.78	-1.9	0.67	71.91	78.32	1.72	0.18	-5.31
SVM		**0.84**	**31.14**	**39.32**	1.96	0.71	**0.37**	**0.84**	**28.35**	**35.00**	1.98	**0.65**	**-0.26**
ARIMA SVM		0.51	182.3	206.60	0.91	0.57	-79	0.67	72.65	78.67	1.78	0.23	-5.28

It can be seen from Figure 7 that the methods we chose for the Hung Yen dataset are relatively good. It can be seen that, in general, all methods give predictable results when compared with actual values. The error is not too much, the most stable is still the KNN method for 72h prediction ability. For the remaining prediction points like 6h, 12h, 24h, 48h and 120h, it shows us a little better side of the method. ARIMA-RF is generally not suitable for forecasting the Red River water level although initial results show good results

(Figure 7). Figure 8 shows us the optimization of each individual model. Thanks to the step of determining the most optimal timestep, it has partly helped the methods achieve relatively high performance.

But again, there are some shortcomings in data segmentation that cause our overall test results to differ from the article*.

Through observing the experimental results (Table 5, Figure 9), we find that ARIMA is still not suitable for predicting Red River water level. In addition to conventional forecasts, it is interesting and important to see whether the combined approach is effective in critical situations, such as forecasting several days before a significant rise in water levels (hint flood peak). Figures 10, 11 and 12 present a visual comparison of all multi-stage water level forecasting methods at Hanoi, Vu Quang and Hung Yen stations.

Figure 10 shows the water level forecast results before the peak (4h on July 24, 2017) at Hanoi station with 1 day (Figure 10a), 3 days (Figure 10b). Again, the evidence from the Figures is that RF and KNN give the best results: the predicted values 1 day ahead and 3 days ahead are close to the actual values. When forecasting 3 days and 5 days in advance, KNN and RF can capture the actual data trend quite well but the forecast error is quite high. Other methods do not work well for all of these cases.

Figure 11 shows the peak water level forecast results (August 21, 2016) at Vu Quang station 12 hours in advance (Figure 11a) and 1 day (Figure 11b). The figure shows that the individual methods can predict peaks 1 and 2 days ahead quite accurately. It still captures the data trend quite well in forecasting the next 3 and 5 days but the error is very high. Meanwhile, all other models failed to predict the peak and failed to capture the data trend.

For Hung Yen data sampled hourly, the highest water level is at 10:00 p.m. on July 23, 2014 (368 cm). We forecast the water level 12 hours before the peak (Figure 12b), 24 hours a day (Figure 12c). Figure 12 again demonstrates that the combined method outperforms the single-component methods. When predicting water levels 6 hours in advance, ARIMA-KNN,ARIMA-SVM,ARIMA-RF produce values closest to the actual value (Fig. 12a). However, when forecasting water levels 12h and 24h in advance, hybrid methods no longer give good results as in the 6h case. Single methods have increased in performance, starting to reach parity with hybrid methods. However, ARIMA KNN ranks second after ARIMA RF but the gap between the prediction errors of the two methods is quite wide. For the Hung Yen data set, when predicting the peak 3 days in advance, only ARIMA RF can capture the trend of the data but the forecast error is relatively large.

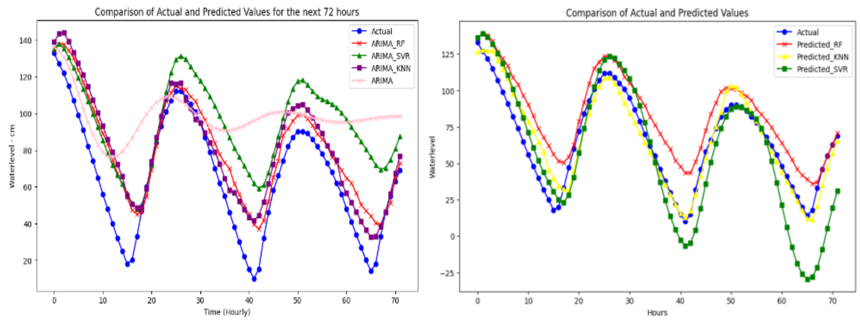


Figure 7: Visual comparison of 72h-ahead predicted values (with hourly frequency) using different forecasting methods with true values on Hung Yen

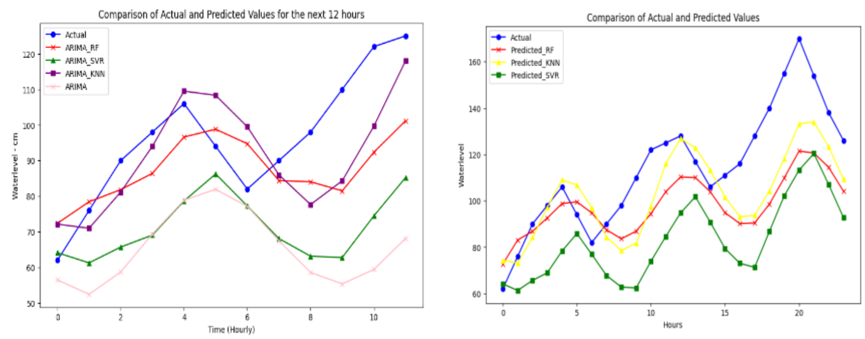
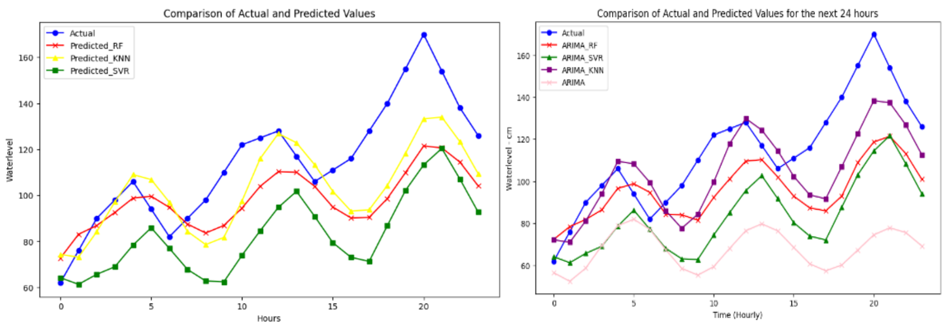


Figure 8: Visual comparison of 10 days-ahead predicted values (with 24 hourly frequency) using different forecasting methods with true values on Hanoi series

5. Conclusion

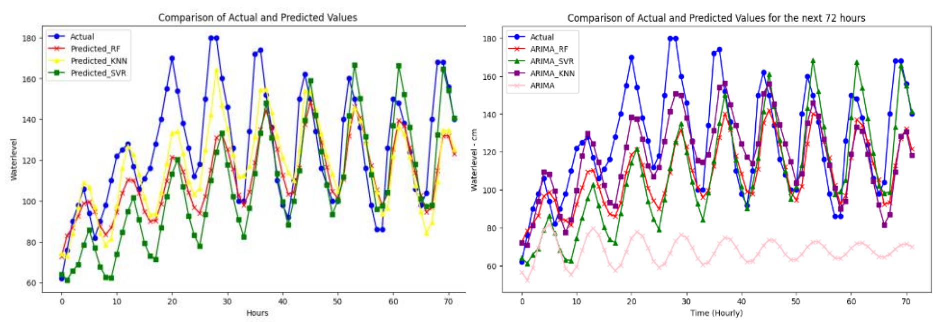
It is a very important but also difficult task to accurately forecast time series, especially the level of water in flood warning systems. In the literature, AIMA, KNN, RF and SVM are widely used and reliable forecasting models that have been applied on a series of hydrological time periods. ARIMA can model linearity well, while other statistical machine learning models are best suited to nonlinear time series.

However, in reality, (hydrological) time series often include both linear and nonlinear correlation structures. Therefore, in this paper, we propose two types of hybrid methods to improve forecasting performance. They took advantage of each individual model type (i.e., linear and nonlinear) and level of complexity (i.e., singly or combined) in time series forecasting. The first type of hybrid method (Model 1) is to combine the original data and residual data obtained from ARIMA to build prediction models, namely ARIMA-KNN, ARIMA-RF and ARIMA-SVR. These models were tested on three large real data sets of Red River water level time series and compared with each individual component model. Test results show that the method combining ARIMA-RF and ARIMA-KNN of Model 1 gives superior and more reliable results than other hybrid models (e.g. ARIMA SVM), as well as better than other methods. Other. Traditional component methods – ARIMA, KNN, RF, SVM. In the future, we plan to apply new methods (ARIMA-RF and ARIMA-KNN) to forecast water levels on other illogical stations of the Red River as well as other rivers.

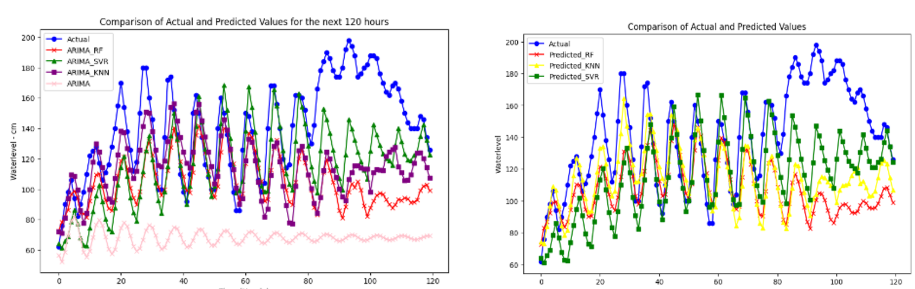


a)

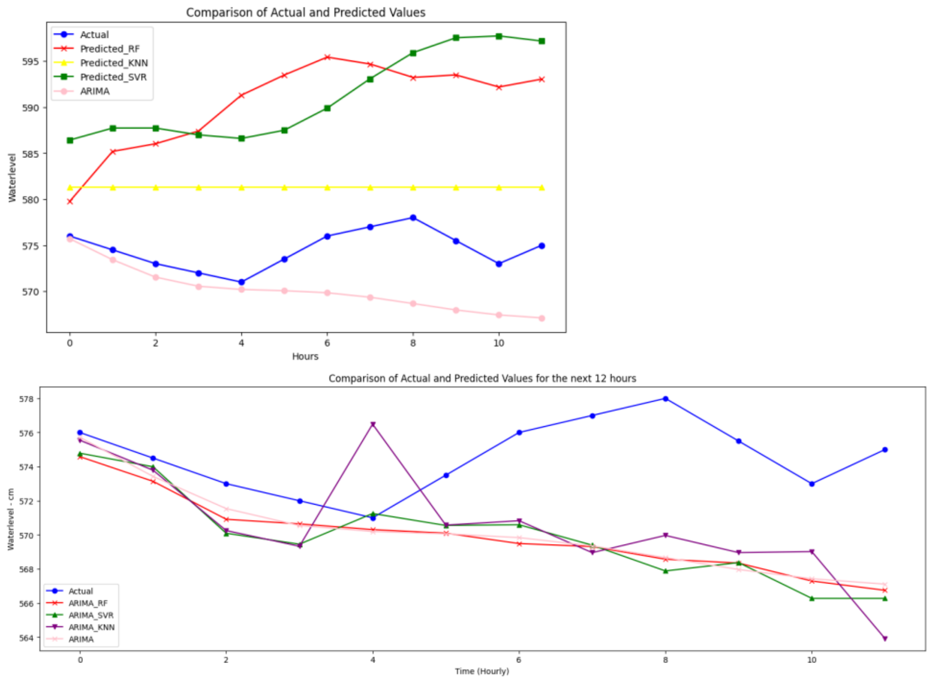
Figure 10: Visual comparison of forecast values generating by different methods on 7/24/2017 at the Hanoi station a) 24h before the peak b) 72h before the peak c) 120h before the peak



b)

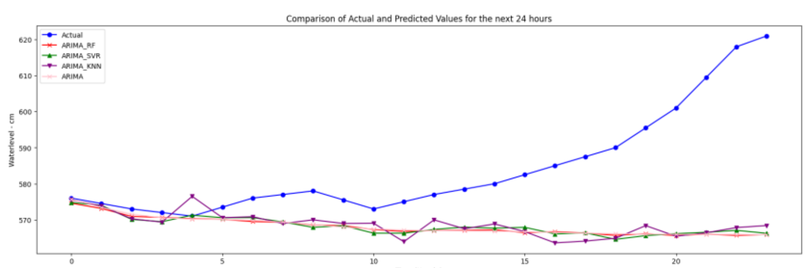


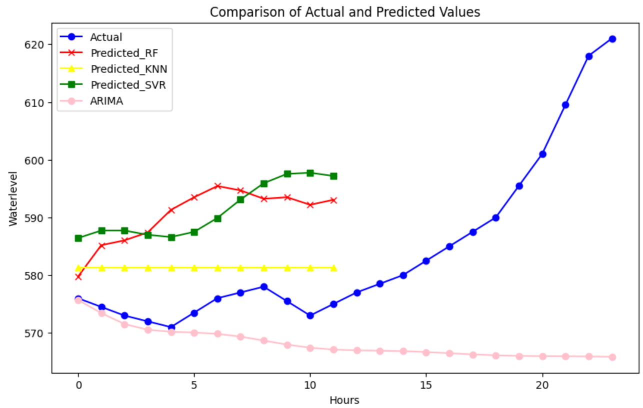
c)



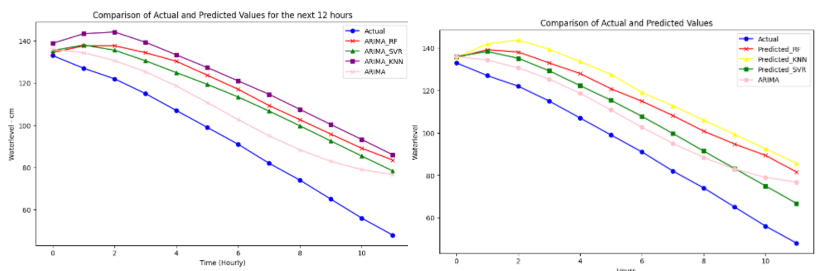
a)

Figure 11: Visual comparison of forecast values generating by different methods on 8/21/2016 at the Vu Quang station a) 12h before the peak; b) 24h before the peak



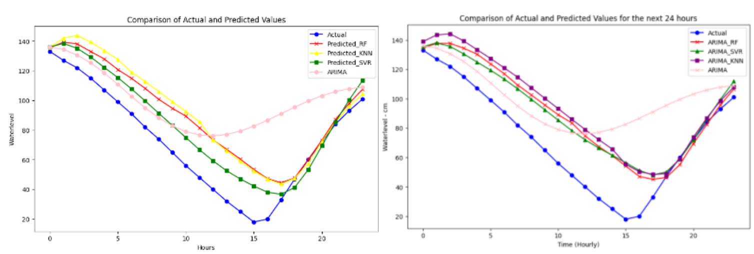


b)



a)

Figure 12: Visual comparison of forecast values generating by different methods on 7/23/2014 at the Hung Yen stationa) 12h before the peak; b) 24h before the peak with hourly frequency



b)

methods (ARIMA_RF and ARIMA_KNN) for forecasting water level on other horological stations of the Red river as well as for other rivers.

Acknowledgments

This research was funded by the national research project titled ”Applications of Artificial Intelligence for Forecasting Hydro-Meteorological Anomalies in Vietnam under the Context of Climate Change”, Grant Number: BDKH.34/16-20

References

Adnan, R.M., Yuan, X., Kisi, O., Adnan, M., Mehmood, A., 2018. Stream Flow Forecasting of Poorly Gauged Mountainous Watershed by Least Square Support Vector Machine, Fuzzy Genetic Algorithm and M5 Model Tree Using Climatic Data from Nearby Station. Water Resources Management 32, 4469–4486.

Aho, K., Derryberry, D., Peterson, T., 2014. Model selection for ecologists: the worldviews of AIC and BIC. Ecology 95, 631–636.

Altman, N.S., 1992. An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician 46, 175–185.

Anderson, P.L., Meerschaert, M.M., Zhang, K., 2013. Forecasting with prediction intervals for periodic autoregressive moving average models: FORECASTING WITH PREDICTION INTERVALS FOR PARMA MODELS. Journal of Time Series Analysis 34, 187–193.

Bafitlhile, T.M., Li, Z., 2019. Applicability of -Support Vector Machine and Artificial Neural Network for Flood Forecasting in Humid, Semi-Humid and Semi-Arid Basins in China. Water 11, 85.

Birylo, M., Rzepecka, Z., Kuczynska-Siehien, J., Nastula, J., 2018. Analysis of water budget prediction accuracy using ARIMA models. Water Science and Technology: Water Supply 18, 819–830.

Box, G., Jenkins, G.M., 1976. Time Series Analysis: Forecasting and Control. Holden-Day.

Breiman, L., 2001. Random forests. Machine Learning 45, 5–32.

Chaari, F., Leskow, J., Napolitano, A., Zimroz, R., Wylomanska, A., Dudek, A. (Eds.), 2015. Cyclostationarity: Theory and Methods - II: Contributions to the 7th Workshop on Cyclostationary Systems And Their Applications, Grodek, Poland, 2014. volume 3 of *Applied Condition Monitoring*. Springer International Publishing, Cham.

Chen, K.Y., Wang, C.H., 2007. A hybrid SARIMA and support vector machines in forecasting the production values of the machinery industry in Taiwan. Expert Systems with Applications 32, 254–264.

Chen, L., Sun, N., Zhou, C., Zhou, J., Zhou, Y., Zhang, J., Zhou, Q., 2018. Flood Forecasting Based on an Improved Extreme Learning Machine Model Combined with the Backtracking Search Optimization Algorithm. Water 10, 1362.

Choi, C., Kim, J., Han, H., Han, D., Kim, H.S., 2019. Development of Water Level Prediction Models Using Machine Learning in Wetlands: A Case Study of Upo Wetland in South Korea. Water 12, 93.

Christopher, O., 2015. Understanding LSTM Networks – colah's blog.

Dang, M., De Smedt, F., 2017. A Combined Hydrological and Hydraulic Model for Flood Prediction in Vietnam Applied to the Huong River Basin as a Test Case Study. Water 9, 879.

Di, C., Yang, X., Wang, X., 2014. A Four-Stage Hybrid Model for Hydrological Time Series Forecasting. PLoS ONE 9, e104663.

Ö mer Faruk, D., 2010. A hybrid neural network and arima model for water quality time series prediction. Eng. Appl. Artif. Intell. 23, 586–594.

Fatichi, S., Vivoni, E.R., Ogden, F.L., Ivanov, V.Y., Mirus, B., Gochis, D., Downer, C.W., Camporese, M., Davison, J.H., Ebel, B., Jones, N., Kim, J., Mascaro, G., Niswonger, R., Restrepo, P., Rigon, R., Shen, C., Sulis, M., Tarboton, D., 2016. An overview of current applications, challenges, and future trends in distributed process-based models in hydrology. Journal of Hydrology 537, 45–60.

Fernandez-Delgado, M., Cernadas, E., Barro, S., Amorim, D., 2014. Do we Need Hundreds of Classiers to Solve Real World Classication Problems? Journal of Machine Learning Research , 3133–3181.

Garcia, F.C.C., Retamar, A.E., Javier, J.C., 2016. Development of a predictive model for on-demand remote river level nowcasting: Case study in Cagayan River Basin, Philippines, in: 2016 IEEE Region 10 Conference (TENCON), IEEE, Singapore. pp. 3275–3279.

Garsole, P., Rajurkar, M.P., 2015. Streamflow Forecasting by using Support Vector Regression, in: 20th International Conference on Hydraulics, Water Resources and River Engineering, India. p. 8.

Ghimire, B.N., 2017. Application of ARIMA Model for River Discharges Analysis. Journal of Nepal Physical Society 4, 27.

Ghumman, A., Ghazaw, Y.M., Sohail, A., Watanabe, K., 2011. Runoff forecasting by artificial neural network and conventional model. Alexandria Engineering Journal 50, 345–350.

Gjika, E., Ferrja, A., Kamberi, A., 2019. A Study on the Efficiency of Hybrid Models in Forecasting Precipitations and Water Inflow Albania Case Study. Advances in Science, Technology and Engineering Systems Journal 4, 302–310.

Hamid, N., Alireza, I., Mahdi, S., Mahdi, A., 2019. Comparing Three Main Methods of Artificial Intelligence in Flood Estimation in Yalphan Catchment. Journal of Geography and Environmental Planning 29.

Haseyama, M., Kitajima, H., 2001. An arma order selection method with fuzzy reasoning. Signal Processing 81, 1331–1335.

Hipni, A., El-shafie, A., Najah, A., Karim, O.A., Hussain, A., Mukhlisin, M., 2013. Daily Forecasting of Dam Water Levels: Comparing a Support Vector Machine (SVM) Model With Adaptive Neuro Fuzzy Inference System (ANFIS). Water Resources Management 27, 3803–3823.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 17351780.

Hrnjica, B., Bonacci, O., 2019. Lake Level Prediction using Feed Forward and Recurrent Neural Networks. Water Resources Management: An International Journal, Published for the European Water Resources Association (EWRA) 33(7), 2471–2484.

Hurvich, C.M., Tsai, C.L., 1989. Regression and time series model selection in small samples. Biometrika 76, 297–307.

Hyndman, R.J., Khandakar, Y., 2008. Automatic time series forecasting: the forecast package for R. Journal of Statistical Software 26, 1–22.

Kasiviswanathan, K., He, J., Sudheer, K., Tay, J.H., 2016. Potential application of wavelet neural network ensemble to forecast streamflow for flood management. Journal of Hydrology 536, 161–173.

Khan, M.S., Coulibaly, P., 2006. Application of Support Vector Machine in Lake Water Level Prediction. Journal of Hydrologic Engineering 11, 199–205.

Khashei, M., Bijari, M., 2010. An artificial neural network (p,d,q) model for timeseries forecasting. Expert Systems with Applications 37, 479–489.

Kim, S.E., Seo, I.W., 2015. Artificial Neural Network ensemble modeling with conjunctive data clustering for water quality prediction in rivers. Journal of Hydro-environment Research 9, 325–339.

Kumar, P., Lohani, A., Nema, A., 2019. Rainfall Runoff Modeling Using MIKE 11 Nam Model. Current World Environment 14, 27–36.

Le, Ho, Lee, Jung, 2019. Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting. Water 11, 1387.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.

Ljung, L., 1986. System Identification: Theory for the User. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Mirzavand, M., Ghazavi, R., 2015. A Stochastic Modelling Technique for Groundwater Level Forecasting in an Arid Environment Using Time Series Methods. Water Resources Management 29, 1315–1328.

Mohan, B.R., Reddy, G.R.M., 2018. A Hybrid ARIMA-ANN Model for Resource Usage Prediction. International Journal of Pure and Applied Mathematics 119, 10.

Mousavi-Mirkalaei, P., Banihabib, M.E., 2019. An ARIMA-NARX hybrid model for forecasting urban water consumption (case study: Tehran metropolis). Urban Water Journal 16, 365–376.

Nash, J.E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models part I A discussion of principles. Journal of Hydrology 10, 282–290.

Nazir, H.M., Hussain, I., Faisal, M., Elashkar, E.E., Shoukry, A.M., 2019. Improving the prediction accuracy of river inflow using two data pre-processing techniques coupled with data-driven model. PeerJ 7.

Nguyen, T.T., Huu, Q.N., Li, M.J., 2015. Forecasting Time Series Water Levels on Mekong River Using Machine Learning Models, in: 2015 Seventh International Conference on Knowledge and Systems Engineering (KSE), IEEE, Ho Chi Minh City, Vietnam. pp. 292–297.

Pannakkong, W., Pham, V.H., Huynh, V.N., 2017. A Novel Hybridization of ARIMA, ANN, and K-Means for Time Series Forecasting:. International Journal of Knowledge and Systems Science 8, 30–53.

Pasupa, K., Jungjareantrat, S., 2016. Water levels forecast in Thailand: A case study of Chao Phraya river, in: 2016 14th International Conference on Control, Automation, Robotics and Vision (ICARCV), IEEE, Phuket, Thailand. pp. 1–6.

Patro, S., Chatterjee, C., Mohanty, S., Singh, R., Raghuwanshi, N.S., 2009. Flood inundation modeling using MIKE FLOOD and remote sensing data. Journal of the Indian Society of Remote Sensing 37, 107–118.

Peng, T., Zhou, J., Zhang, C., Fu, W., 2017. Streamflow Forecasting Using Empirical Wavelet Transform and Artificial Neural Networks. Water 9, 406.

Phan, T.T.H., Poisson Caillault, ., Lefebvre, A., Bigand, A., 2017. Dynamic time warping-based imputation for univariate time series data. Pattern Recognition Letters .

Rezaie-Balf, M., Kim, S., Fallah, H., Alaghmand, S., 2019. Daily river flow forecasting using ensemble empirical mode decomposition based heuristic regression models: Application on the perennial rivers in Iran and South Korea. Journal of Hydrology 572, 470–485.

Riad, S., Mania, J., Bouchaou, L., Najjar, Y., 2004. Rainfall-runoff model usingan artificial neural network approach. Mathematical and Computer Modelling 40, 839–846.

Seo, Y., Kim, S., Kisi, O., Singh, V.P., 2015. Daily water level forecasting using wavelet decomposition and artificial intelligence techniques. Journal of Hydrology 520, 224–243.

Shibata, R., 1976. Selection of the order of an autoregressive model by Akaike's information criterion. Biometrika 63, 117–126.

Temr, A.S., Akgn, M., Temr, G., 2019. Predicting Housing Sales In Turkey Using Arima, LSTM And Hybrid Models. Journal of Business Economics and Management 20, 920–938.

Toro, C.H.F., Gmez Meire, S., Glvez, J.F., Fdez-Riverola, F., 2013. A hybrid artificial intelligence model for river flow forecasting. Applied Soft Computing 13, 3449–3458.

Tran Anh, D., Hoang, L.P., Bui, M.D., Rutschmann, P., 2018. Simulating Future Flows and Salinity Intrusion Using Combined One- and Two-Dimensional Hydrodynamic ModellingThe Case of Hau River, Vietnamese Mekong Delta. Water 10, 897.

Valenzuela, O., Rojas, I., Rojas, F., Pomares, H., Herrera, L., Guillen, A., Marquez, L., Pasadas, M., 2008. Hybridization of intelligent techniques and ARIMA models for time series prediction. Fuzzy Sets and Systems 159, 821–845.

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.

Vapnik, V.N., 1995. The Nature of Statistical Learning Theory. Springer-Verlag, Berlin, Heidelberg.

Wang, X., Liu, T., Zheng, X., Peng, H., Xin, J., Zhang, B., 2018. Short-term prediction of groundwater level using improved random forest regression with a combination of random features. Applied Water Science 8, 125.

Wongsathan, R., Jaroenwiriyapap, W., 2016. A hybrid ARIMA and RBF neural network model for tourist quantity forecasting: a case study for Chiangmai province. KKU Research Journal 21(1), 37–54.

Woodward, W.A., Gray, H.L., Elliott, A.C., 2017. Applied time series analysis, with R. Second edition ed., CRC Press, Taylor & Francis Group, Boca Raton.

Wu, C., Chau, K., Li, Y., 2009. Methods to improve neural network performance in daily flows prediction. Journal of Hydrology 372, 80–93.

Xie, Y., Lou, Y., 2019. Hydrological Time Series Prediction by ARIMA-SVR Combined Model based on Wavelet Transform, in: Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence - ICIAI 2019, ACM Press, Suzhou, China. pp. 243–247.

Xu, G., Cheng, Y., Liu, F., Ping, P., Sun, J., 2019. A Water Level Prediction Model Based on ARIMA-RNN, in: 2019 IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService), IEEE, Newark, CA, USA. pp. 221–226.

Yan, H., Zou, Z., 2013. Application of a Hybrid ARIMA and Neural Network Model to Water Quality Time Series Forecasting. Journal of Convergence Information Technology(JCIT) 8.

Yang, J.H., Cheng, C.H., Chan, C.P., 2017. A Time-Series Water Level Forecasting Model Based on Imputation and Variable Selection Method. Computational Intelligence and Neuroscience .

Yaseen, Z.M., Fu, M., Wang, C., Mohtar, W.H.M.W., Deo, R.C., El-shafie, A., 2018. Application of the Hybrid Artificial Neural Network Coupled with Rolling Mechanism and Grey Model Algorithms for Streamflow Forecasting Over Multiple Time Horizons. Water Resources Management 32, 1883–1899.

Yu, Z., Lei, G., Jiang, Z., Liu, F., 2017. ARIMA modelling and forecasting of water level in the middle reach of the Yangtze River, in: 2017 4th International Conference on Transportation Information and Safety (ICTIS), IEEE, Banff, AB, Canada. pp. 172–177.

Zhang, G.P., 2003. Time series forecasting using a hybrid arima and neural network model. Neurocomputing 50, 159–175.

Zhong, C., Guo, T., Jiang, Z., Liu, X., Chu, X., 2017a. A hybrid model for water level forecasting: A case study of Wuhan station, in: 2017 4th International Conference on Transportation Information and Safety (ICTIS), IEEE, Banff, AB, Canada. pp. 247–251.

Zhong, C., Jiang, Z., Chu, X., Guo, T., Wen, Q., 2017b. Water level forecasting using a hybrid algorithm of artificial neural networks and local Kalman filtering. Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment 233, 174–185.

Zhu, C., Liang, Q., Yan, F., Hao, W., 2013. Reduction of Waste Water in Erhai Lake Based on MIKE21 Hydrodynamic and Water Quality Model. The Scientific World Journal 2013, 1–9.