FACTA UNIVERSITATIS
Series: **Automatic Control and Robotics** Vol. 23, Nº 1, 2024, pp. 77 - 94
https://doi.org/10.22190/FUACR240111002M

Survey Paper

DAILY DANUBE RIVER WATER LEVEL PREDICTION USING EXTREME LEARNING MACHINE APPROACH

UDC (004.85)

Miljana Milić[1], Novak Radivojević[1], Jelena Milojković[1], Miljan Jeremić[2]

[1]University of Niš, Faculty of Electronic Engineering, Department of Electronics, Republic of Serbia
[2]University of Niš, Faculty of Civil Engineering, Department of Mathematics, Physics and Informatics, Republic of Serbia

ORCID iDs: Miljana Milić https://orcid.org/0000-0001-7037-7709
Novak Radivojević https://orcid.org/0009-0000-4423-6224
Jelena Milojković N/A
Miljan Jeremić N/A

Abstract. *Anticipating water levels in vast riverbeds is crucial for preventing and mitigating floods or droughts, assessing power plant capacity, and facilitating navigation management. This study introduces an innovative water level prediction model utilizing an Extreme Learning Machine developed to solve the issues of low performance of existing forecasting methods. Development of such a system is of extreme importance when talking about the largest European river – the Danube River. Experimental findings reveal the model's satisfactory performance across various accuracy metrics, complexity considerations, and calculation speed. The prediction with the highest error rate based on MAPE criteria was for Prahovo water level prediction over a 365-day period at 2.02%, whilst the most accurate predictions were for Novi Sad and Banatska Palanka over 30 days and 180 days horizons, respectively, at 0.0550%. The highest coefficient of determination (R2) was achieved with the Novi Sad data at 0.9968, whilst the lowest was observed with the Prahovo data at 0.7353. The ELM model achieved high precision by adjusting the activation functions of the hidden layer neurons, which involved using different combinations of sigmoid and radial-basis activation functions.*

Key words: *Extreme learning machine, Danube river level, time series, forecasting.*

Received January 11, 2024 / Accepted March 04, 2024
Corresponding author: Miljana Milić
University of Niš, Faculty of Electronic Engineering, Aleksandra Medvedeva 4, 18000 Niš, Republic of Serbia
E-mail: miljana.milic@elfak.ni.ac.rs

1. INTRODUCTION

Water is one of the most precious natural resources we possess [1]. Without it, life on Earth would not exist. The laws of nature limit our access to water. Although there is an abundance of water on Earth, it is not always easily accessible when needed, at the correct place, or of the right quality. Chemical pollutants that were improperly disposed earlier are now showing up in our water systems. The study of hydrology has evolved over time in order to better understand the complex water systems of the Earth and help with water-related problems. Hydrological challenges require innovative solutions.

On the other hand, it is widely acknowledged that the Danube River provides the foundation for the interstate collaboration and economic growth of the Danube countries. However, the Danube has a significant role in other areas as well, like the promotion of general cooperation among the Danube countries and socioeconomic, cultural, and political development. The Danube River plays a major role in planning and development of appropriate ecological concepts in the sphere of environmental protection, as well as in other areas of interstate cooperation involving the entire Danube region.

Reliable forecasts for various forecast horizons are necessary for solving critical decision-making problems. The hydrological characteristics of the Danube River impact a number of concerns, including navigation, droughts and floods, power plant capacity, and other issues. In time series research, forecasts have traditionally been made using a wide variety of statistical methods, including autoregressive models (ARIMA, SARIMA, etc.), exponential smoothing, dynamic regression models and many others [2-7]. Along with these methods, deep learning models have become popular in research for solving time-series prediction challenges due to their ease of implementation, availability of tools to create prediction models and notable achievements in many different areas of application [8-10]. Deep learning techniques encompass various artificial neural network structures, where recurrent networks such as Long Short-Term Memory (LSTM) [11], [12], Gated Recurrent Units (GRU), etc. stand out in the time series prediction domain due to their ability to capture time series patterns from historical data [13], [14]. The issue that arises with deep learning models is the lengthy process of network training when dealing with a considerable amount of data, together with the complex iterative computation of network parameters. This is particularly an issue with recurrent models as their training process causes high memory consumption due to the models' memory effect and due to the use of backpropagation through the time algorithm for updating the network weights. Another approach, the Extreme Learning Model (ELM), has arisen as a response to the expensive nature of neural network training. Unlike typical neural networks, it does not utilize an iterative error function minimization strategy. In this algorithm, a subset of network weights is initialized to a random value and during training, the rest of the weights are adjusted according to the training input-output pairs using a much simpler process. Due to its speed, simplicity and increasingly widespread usage, ELM is chosen to be the algorithm implemented in this paper for predicting the water level of Danube on multiple measurement stations.

A number of researchers address the hydrological forecasting challenges of the Danube River by employing different statistical or neural network modeling methodologies [15-19]. Because hydrological time series are influenced by numerous independent variables, traditional forecast models struggle to provide effective forecasts. Some advanced hybrid forecasting models have recently been created in order to improve reliability and precision of forecasting results [19-24].

In the study presented in this article, a development of a specific Extreme Learning Machine (ELM) model is performed to make one-step-ahead predictions for daily-time horizons in the Danube River flow through the Republic of Serbia, at seven measuring stations. They are: Bezdan, Zemun, Novi Sad, Banatska Palanka, Veliko Gradište, Donji Milanovac, and Prahovo. The initial study in this field was conducted in [25]. Further research, performed in this study aims to assess and improve the accuracy of the forecasts. Brief descriptions of water level data and ELM modeling follow. After that the Python computer language implementation of the forecasting model is shown. At the conclusion, appropriate forecast performance measures are evaluated and discussed together with suggestions for future research.

2. Study Area

The Danube River's natural attributes set it apart and differentiate it from other European rivers. This river is Europe's watershed due to its length of 2,888 km, navigability, rich natural content, fish, and plant life. Geographically, the Danube springs in the Schwartzwald Mountains of Baden-Württemberg, which are located in the southwest of the Federal Republic of Germany. It is formed by the merger of the smaller rivers Briga and Breg near Donaueschingen [26]. The Danube flows from west to east, passing through several major towns in Central and Eastern Europe (Vienna, Bratislava, Budapest, and Belgrade) before forming a delta in Romania and Ukraine after 2850 kilometers on the Black Sea coast. Throughout history, the Danube has always been an important international waterway.

The Danube is the second European river in length and with a long-term daily mean discharge of about 6500 m^3/s [27]. Fig. 1 shows the portion of the flow of the Danube that goes through Serbia. A section of its course forms a natural border between Serbia and Croatia, as well as Serbia and Romania (138 km toward Croatia and 227 km toward Romania). The Danube has an average width of 600 meters and a maximum width of 2000 meters at the entrance to the Đerdap Gorge. The Đerdap Gorge, which is 97 kilometers long, is Serbia's largest gorge.

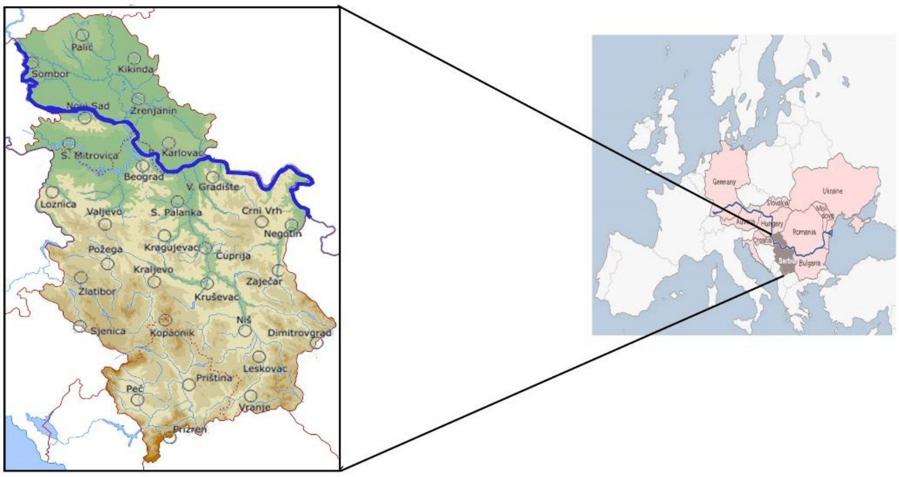


Fig. 1 The Danube River through the Republic of Serbia

The Danube basin, which covers an area of around 801,463 km^2 and represents 10% of the area of the European continent, is home to approximately 80.5 million people in 19 countries through which this river flows [27-29]. The Danube basin is separated into four areas based on its geological structure and geographical layout: upper, middle, and lower Danube, as well as the Danube's delta [30]:

1. Upper Danube Region, between its springs and the Devin Gate, (1,880 r km (river kilometer), basin area: 131,338km^2, long-term annual average discharge 2,051m^3/s).
2. The Central Danube Region lays between the Devin Gate and the Iron Gate (930 r km, basin area 444,894km^2, long-term annual average discharge 5,585m^3/s).
3. The Lower Danube Region is placed between the Iron Gate and the Danube's Delta (132 r km, basin area: 230,768 km^2, long-term annual average discharge 6,563m^3/s).
4. The Danube Delta is located at the Black Sea coast. As a crucial wetland in the Danube River Basin, the Danube Delta covers 6,750 km^2 in area [31], [32].

Since the Danube River supplies water for industry, agriculture, and several eco-systems, it is generally of tremendous ecological, social, and economic worth. In addition, the Danube is crucial for transportation, electricity production, recreation, tourism, fishing, and biodiversity. Aside from its natural features, the Danube region can be recognized by a variety of other characteristics, such as historical legacy, future-oriented planning, increased economic activity, collaboration in various areas of social life, etc.

In 2014 the Republic of Serbia's territory experienced one of the worst natural disasters i.e., an extreme flood [33]. The floods preventions systems either failed to prevent damage, or were never implemented. The riverbeds were rarely cleaned and maintained for decades, and the embankments were not renovated. The pumping stations were also neglected and their functionality was rarely checked. The authorities of the Republic of Serbia have conducted a post-disaster damage assessment after the floods [34], [35]. In general, the serious unpreparedness disturbed lives of many people and animals and did severe damages in urban systems, agriculture, and local economy [36].

Considering the noted deficiencies and the profound impact on the community, there is a compelling need and a motive to proactively address and enhance flood prediction and prevention measures especially in the context of such a large river basin like the Danube's basin. Current state of readiness, as evidenced by the 2014 flood, is insufficient to mitigate the potential future threats posed by the Danube River. The study, conducted in this research aims to contribute to the scientific understanding and practical management of water-related disasters. By employing advanced predictive modeling techniques, such as the ELM approach, this research attempts to provide accurate and timely predictions of the Danube River's water levels. The ultimate goal is to develop a robust framework that can assist authorities in the Republic of Serbia and other regions prone to flooding in implementing more effective flood prevention strategies.

3. The Methodology of ELM Modelling

Extreme Learning Machine (ELM) is a feed-forward neural network (FFNN) training algorithm based on randomization of network weights, which is primarily applied to networks with a single hidden layer. ELM aims to achieve very high learning speed by initializing a portion of network parameters to a random value using a distribution function, rather than implementing an iterative process and backpropagation to update the weights' values [37]. In particular, weights between the input layer and the hidden layer, as well as the bias of the hidden layer are randomized, while the weights between the hidden layer and the output are determined in a single step by solving a system of linear equations. This method is shown to be reliable through rigorous mathematical proofs and strong definitions [37].

Since no iterative steps are required, ELM's weight randomization has shown to be a much faster method compared to other algorithms used extensively (such as Gradient Descent) while keeping accuracy at a high level. Short training times achieved by the algorithm combined with easy implementation of feed-forward NNs have contributed to ELM being widely used in many different fields of application.

To generalize, Extreme Learning Machine time series modelling offers several advantages over other forecasting methodologies:

- Fast Training: ELMs usually consist of a single layer of hidden neurons that are randomly initialized, resulting in much quicker training when compared to conventional neural networks such as feedforward or recurrent neural networks. This enables faster testing and refinement of models [38].
- Simple implementation: ELMs are easy to implement and need minimal hyperparameter adjustment. Due to the single layer structure, they have a fewer number of parameters to optimize in comparison to other neural network topologies, which makes them simpler to train and implement.
- Efficient nonlinear modelling: ELMs have demonstrated the ability to perform as well as more advanced time series forecasting techniques, using fewer computer resources. Time series data with nonlinear relationships can be captured by ELMs without the need for explicit feature engineering. They automatically extract pertinent aspects from the incoming data, enabling them to efficiently represent intricate patterns and dynamics. This efficiency makes them appropriate for real-time forecasting applications or situations with restricted processing resources [39], [40].
- Generalization capability: ELMs have shown strong generalization capacity, especially when working with noisy or high-dimensional time series data. When applied to properly pre-processed data, overfitting is less likely to occur [41].
- Scalability: ELMs are capable of efficiently processing large-scale time series datasets because of their straightforward and parallelizable training procedure. Their scalability makes them well-suited for applications requiring handling large volumes of data, including financial forecasting or energy demand prediction [42].

A. *Mathematical formulation of the ELM model*

Mathematical model of the ELM algorithm will be explained on an example regression network (Fig. 2) with L hidden nodes, where:

- x_i is the i-th input node for input data of length N,
- $\mathbf{W}$ is the weight matrix of connections between the input and the hidden layer,
- b_i is the i-th hidden layer bias coefficient,
- a_i is the activation of i-th hidden node,
- y is the output node,
- β_i is the weight of a connection between the i-th hidden neuron and the output.

The goal of the ELM training algorithm is to find the optimal values of the output weights $\beta_1, \beta_2, \dots \beta_L$ in order to find the best fit model based on the training data. The training algorithm consists of the following steps:

1) Training data preparation
2) Weight and bias initialization
3) Hidden layer output matrix calculation - **H**
4) Calculation of $\mathbf{H}^{\dagger}$ (the Moore-Penrose inverse of **H**)
5) Output weights vector calculation

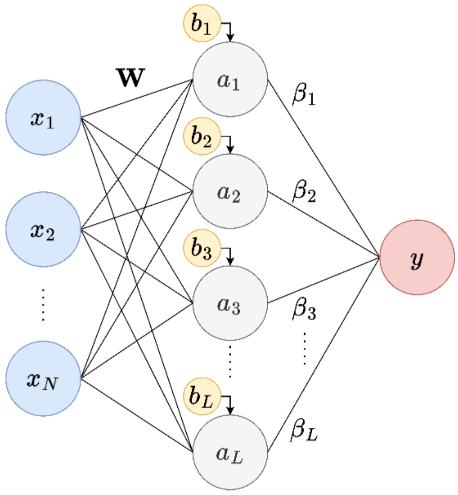


Fig. 2 Neural network representation of ELM

Before the training, the pre-processed data set is split into training and test sets. The sets consist of pairs of input vectors and corresponding outputs. The portion of the data set used for training is usually picked to be 60% - 80% of the entire dataset.

Firstly, the input weights in matrix **W** and the biases of the hidden layer are assigned random values with a selected probability distribution. Then, the training samples are passed to the network one by one. For each input-output training pair, the activations of the hidden neurons are calculated by applying an arbitrary activation function $g(x)$. For the training sample t, the activations are computed as specified in equation (1):

$$a_i^{(t)} = g(\boldsymbol{w}_i \boldsymbol{x}^{(t)} + b_i) \tag{1}$$

Here, $\mathbf{w}_i$ is the vector of weights connecting i-th hidden node and the inputs, and $\mathbf{x}^{(t)}$ is the t-th input vector of the training set. All the activations are then stored in the hidden layer output matrix **H**, as shown in equation (2). The matrix consists of L columns (one for each hidden neuron) and n rows where i-th row of the matrix contains activations calculated for the i-th training pair.

$$\boldsymbol{H} = \begin{bmatrix} a_1^{(1)} & a_2^{(1)} & \cdots & a_L^{(1)} \\ a_1^{(2)} & a_2^{(2)} & \cdots & a_L^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{(n)} & a_2^{(n)} & \cdots & a_L^{(n)} \end{bmatrix} \tag{2}$$

According to the model, the network output can be determined as a weighted sum of hidden node activations with $\boldsymbol{\beta}$ being the output weights vector. If all the output data in the training set are stored in a vector $\boldsymbol{y}$, then it follows that $\boldsymbol{H\beta} = \boldsymbol{y}$ which represents a system of linear equations in output weight coefficients consisting of *n* equations and *L* unknowns. The solutions to $\boldsymbol{H\beta} = \boldsymbol{y}$ can only be approximate since the high number of training samples results in a system with many more equations than variables which is impossible to solve directly. An approximate solution can be selected so that the norm of the system is minimized, i.e.:

$$\left\|\boldsymbol{H}\hat{\boldsymbol{\beta}} - \boldsymbol{y}\right\| = \min_{\boldsymbol{\beta}} \|\boldsymbol{H\beta} - \boldsymbol{y}\| \tag{3}$$

In other words, the resulting vector of $\boldsymbol{H\beta}$ is as close as possible to $\boldsymbol{y}$, which results in a least-squares solution. It can be shown that the solution which satisfies the minimum norm condition is in the form of [43]:

$$\hat{\boldsymbol{\beta}} = \boldsymbol{H}^{\dagger}\boldsymbol{y} \tag{4}$$

Where $\boldsymbol{H}^{\dagger}$ represents the Moore-Penrose pseudoinverse matrix of $\boldsymbol{H}$, which can be calculated as follows:

$$\mathbf{H}^{\dagger} = (\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}} \tag{5}$$

In this way, the network output weights are determined directly using a single matrix expression and calculating the Moore-Penrose pseudoinverse could be performed with the help of simpler algorithms such as singular value decomposition. After determining the output weights, the model is completely trained and can be tested or used for making predictions.

B. *ELM model fitting and prediction accuracy measures*

Some of the metrics commonly used for evaluating the accuracy of neural network prediction are Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R^2 or the coefficient of determination and many others. In this paper, MAPE, RMSE and R^2 were used for evaluating the models.

MAPE quantifies the accuracy of predicted values by calculating the average of the absolute percentage errors over all the predictions. The use of MAPE is often suitable for evaluating predictions made for huge datasets. MAPE can be calculated as follows:

$$MAPE = \left(\frac{1}{n}\sum_{i=1}^{n}\left|\frac{p_i - y_i}{y_i}\right|\right) \cdot 100\%, \tag{6}$$

where p_i is the value predicted by the model, y_i is the actual value from the dataset and n is the number of values in the dataset.

RMSE is another metric commonly employed in regression or time series model training. It quantifies the deviation of the predicted data from the line of best fit. It can serve as a decisive factor for choosing the most effective forecasting model from a set of models trained on the same dataset. RMSE can be calculated as follows:

$$RMSE = \sqrt{\frac{1}{n}\left(\sum_{i=1}^{n}(p_i - y_i)^2\right)}. \tag{7}$$

R^2, also known as the coefficient of determination, is a statistical metric used in regression analysis to quantify the percentage of variance in the dependent variable that can be predicted by the independent variable. R^2 quantifies the degree to which the regression model fits the dataset. R^2 can be calculated as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - p_i)^2}{\sum_{i=1}^{n}(y_i - \hat{y})^2} \quad (8)$$

where $\hat{y}$ is the mean value of the dataset. A model can be considered as a good fit if its R^2 value is close to 1.

4. Danube Water Level Data

The source of the data used for creating ELM water level prediction models comes from the hydrology section of the online-accessible weather records of Republic Hydrometeorological Service of Serbia (RHSOS) which contains measurements of a variety of hydrological characteristics of rivers in Serbia [44]. These characteristics include daily water level, water temperature, water flow measurements, ice phenomena, quality of water, and more.

The water level data provided by RHSOS is collected from the network of measurement stations installed on various locations including many rivers and lakes in Serbia. As of 2023, a total of 211 such stations are located throughout the territory of Serbia, with 15 of those stations being located on the Danube River [45]. There are a number of instruments that are utilized at the stations in order to measure the water level such as limnigraph water gauges and/or digital devices. At each station, measuring of the water level takes place once a day at a particular hour, on a regular basis. Water level is measured in centimeters and relative to a zero-elevation point defined for each station. Zero-elevation points are given in meters above the Adriatic Sea.

Every year, in the middle of the year, RHSOS publishes an annual report that includes information on the surface water parameters that were measured during the previous year. The section of the annual report regarding water level includes daily measurements for all stations, as well as minimum, average, and maximum readings for each month and for the entire year.

The ELM model for predicting the Danube water level was developed based on data from several RHSOS annual reports. Out of the total of 15 stations, 7 were selected: Bezdan, Novi Sad, Zemun, Banatska Palanka, Veliko Gradište, Donji Milanovac, and Prahovo. The RHSOS online source offers reports starting from 1991 up to 2021 (at the time of developing the models). However, for some stations, data before a particular year is missing and only a subset of the annual reports was used per station, starting from the year when the data was recorded for the first time for the given station and up to 2021 Fig. 3 provides a graphical representation of each of the stations' datasets, where water level values are shown relative to the Adriatic Sea level i.e. including the zero-elevation points of each station. Fig. 4 shows the distribution of water levels appearing in the data set relative to zero-elevation points. Table 1 presents a summary of station statistical details as well as the starting year of the subset of annual reports used for a particular dataset. Table 2 shows a sample of one of the datasets.

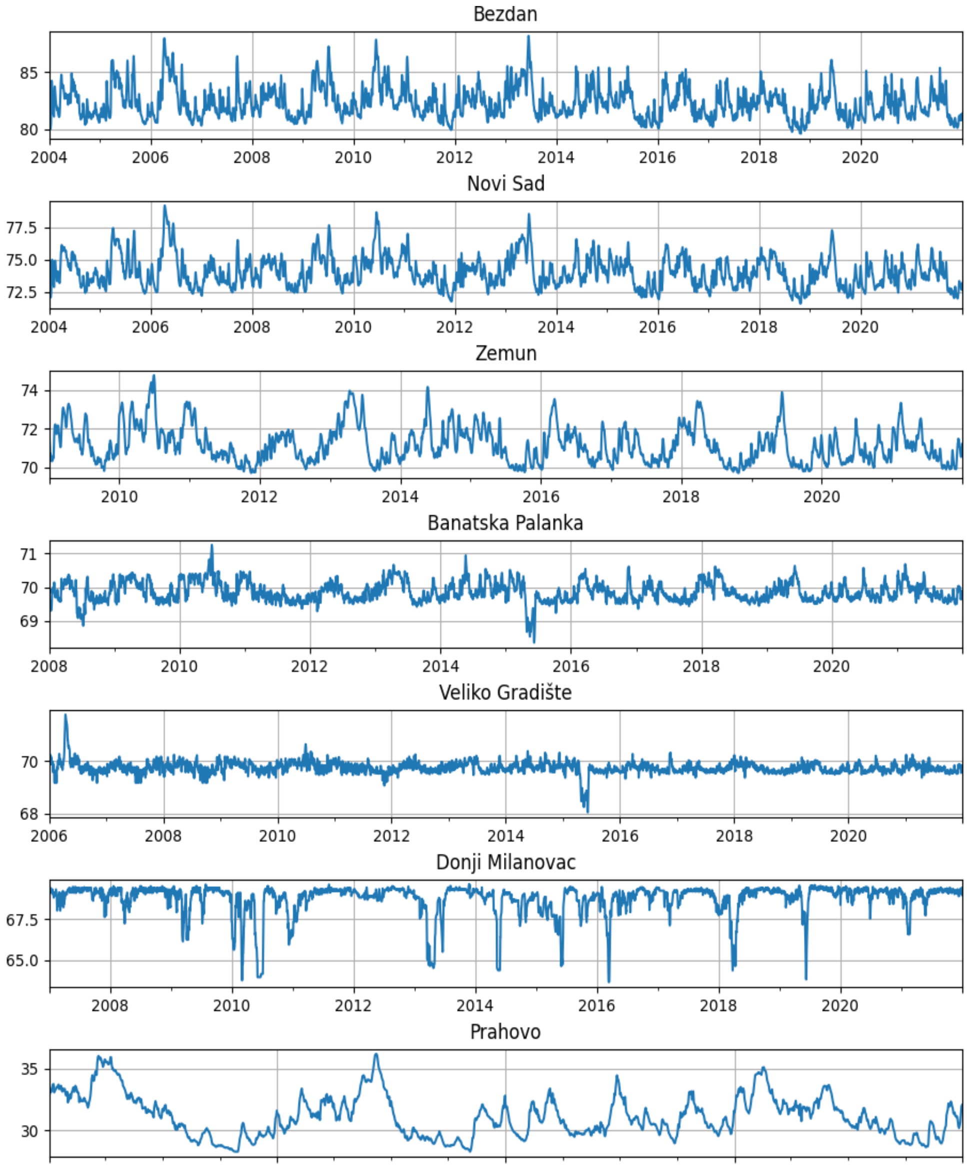


Fig. 3 Water level data measurements from each station

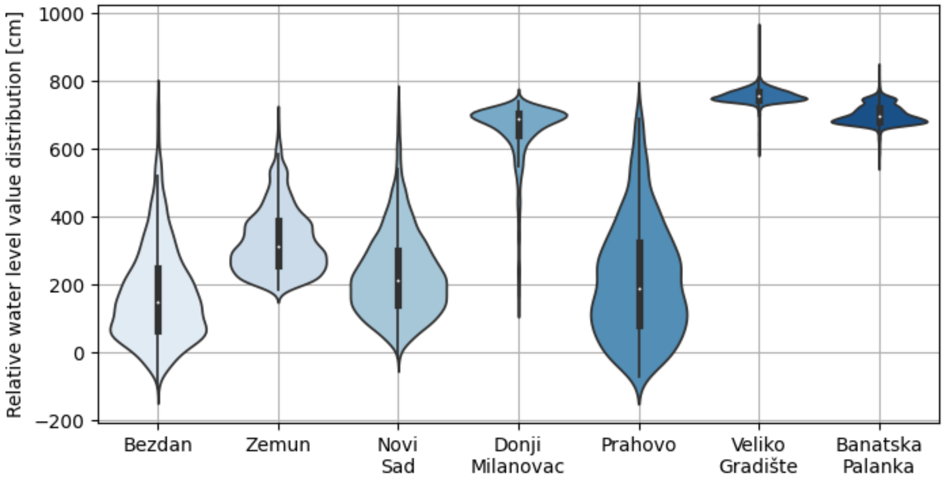


Fig. 4 Water level distribution violin plots across stations

Table 1 Statistical details of different datasets

	Bezdan	Novi Sad	Zemun	Donji Milanovac	Veliko Gradište	Banatska Palanka	Prahovo
Zero-elevation [m]	80.64	71.73	67.87	62.24	62.17	62.85	29.02
Starting year	2004	2004	2009	2007	2006	2008	2018
No. of samples	6575	6575	4748	5479	5844	5114	1461
Mean relative water level [cm]	171.64	230.49	331.28	651.34	757.64	701.14	214.18
Std. deviation	139.4	121.35	95.51	97.44	23.01	29.75	168.41
Min. [cm]	-101	-13	184	140	588	551	-72
25% (Q1) [cm]	166	141	256	642	745	680	80
50% (Q2) [cm]	148	215	312	687	755	695	190
75% (Q3) [cm]	248	302	388	705	770	721	324
Max. [cm]	755	744	691	742	960	840	718

For each dataset, the data from the reports has been organized into a single time series with samples ranging from a starting date to 31.12.2021. and then stored in individual *.csv* files. Missing data was imputed using linear interpolation. In order to eliminate unwanted noise in the data, a convolution filter (Centered Moving Average) was employed on each individual dataset.

5. Development and Training of ELM Models

Prediction of the next sample in a time series is in most cases based on the samples collected in a number of previous consecutive timesteps. For this purpose, the data is firstly reorganized into input-output pairs that are appropriate for time series prediction before the ELM models are trained. The sliding window technique is used to generate vector pairs for each model. This involves creating input-output pairs from consecutive dataset points of each dataset. For a model with an input length of N, the i-th pair is formed by taking in total N+1 consecutive points, with points x_i, x_{i+1}, x_{i+2}, … x_{i+N-1}, forming the input vector of the pair and taking x_{i+N} as the target or the desired output. The total number of input-output pairs generated in this way is n-N-1, where n is the length of the dataset used for training. In this paper, 70% of all reorganized data in each dataset was used for training. The accuracy of the forecast can be influenced by the length of the input vector, also known as the prediction horizon, depending on the time series [46]. This study involved developing models for each station with input lengths taking 30, 60, 120, 180, and 365 samples and then selecting the model that yields the most accurate predictions after training. This accounts for a total of 35 models, one per horizon length and per dataset.

The structure of each ELM model is selected using the method of optimal pruning described in detail in [47]. This method involves selecting an initial number of hidden neurons, grading the neurons according to their “usefulness” (using RMSR [48]), then performing a Leave-One-Out (LOO) cross-validation to evaluate the model performance and, in the end, discarding some of the neurons based on their grade and the results of LOO. For models in this study, hidden layers were initially assigned 600 neurons in total, 100 of them implementing the sigmoid activation function and the other 500 the radial basis function with L2 norm, which is commonly used in ELMs [49].

6. Results and Discussion

After training all the models, prediction testing was conducted using the corresponding test sets, which consisted of the remaining 30% of the data set that was not used for training. Table 3 shows the accuracy metrics for each station across different horizons, together with the count of hidden neurons in each network following optimal pruning. Fig. 5 shows the graphical representation of predictions with the highest R^2 value among different horizons over the entire test set for each station. Additionally, the subset of 60 consecutive predictions for which each stated model had the lowest RMSE is shown. Predicted values take the zero-elevation level into account. Plot lines colored in blue represent the actual recorded values, while red lines represent predictions.

Given the results in the tables, it can be concluded that the methodology used for generating and training has yielded models that perform very well across nearly all datasets. Most of the models obtained an R^2 value larger than 0.95, except for the model for the Veliko Gradište station which resulted in R^2 around 0.86 across all horizons due to larger levels of noise in the data than in the other datasets, but had considerably low RMSE values on the other hand. Among all the datasets, the Prahovo station had the smallest number of samples and thus the small training and test sets, which explains the drop in R^2 as the horizon increases. It can also be noticed that the longer the horizon, the less neurons would be dropped out of the network after pruning. Overall, the models were able to closely match the trend and predict sudden rises/drops of the water level, regardless of the water level values distribution and the value of the zero-elevation point.

Table 2 Summary of prediction accuracy and network complexity according to different criteria – a) Bezdan model, b) Zemun model, c) Novi Sad, d) Veliko Gradište model, e) Donji Milanovac model, f) Banatska Palanka model, g) Prahovo model

Bezdan	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.0759	9.5251	0.9932	30	160
60	0.0784	9.7075	0.9930	48	268
120	0.0858	10.2512	0.9922	81	433
180	0.0888	10.4440	0.9920	92	479
365	0.1154	12.8260	0.9881	100	500

Zemun	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.0556	5.2561	0.9961	21	114
60	0.0579	5.4199	0.9959	47	206
120	0.0604	5.6002	0.9956	60	334
180	0.0705	6.3614	0.9939	99	485
365	0.0817	7.3589	0.9907	100	500

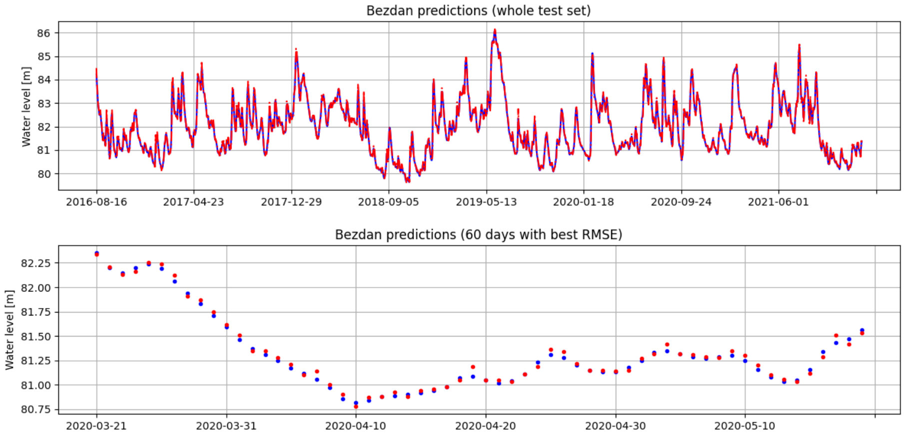
Novi Sad	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.0550	5.6510	0.9968	28	138
60	0.0563	5.7810	0.9967	39	218
120	0.0604	6.1031	0.9963	78	388
180	0.0619	6.1402	0.9963	100	495
365	0.0832	7.9453	0.9939	100	500

Veliko Gradište	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.0562	5.1765	0.8642	8	48
60	0.0558	5.1451	0.8653	39	171
120	0.0558	5.1348	0.8667	39	217
180	0.0553	5.0975	0.8679	62	265
365	0.0561	5.1336	0.8678	99	499

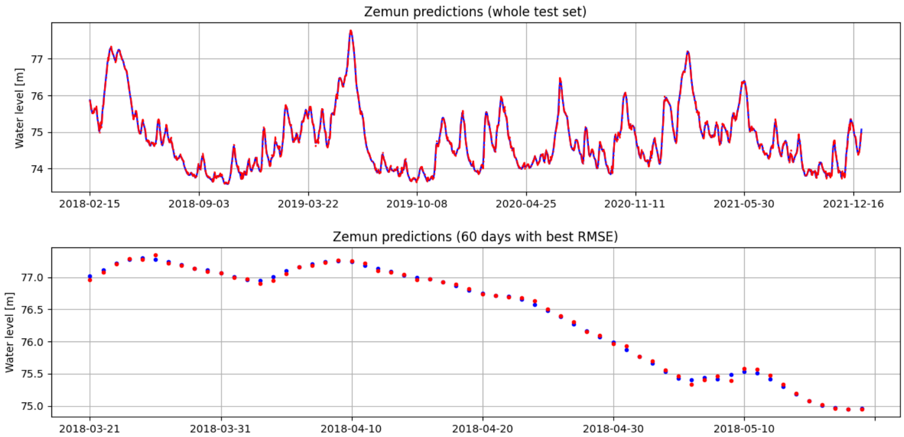
D. Milanovac	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.1202	11.9476	0.9747	25	99
60	0.1213	11.9684	0.9747	51	238
120	0.1226	12.0287	0.9747	89	445
180	0.1245	12.1552	0.9744	100	500
365	0.1301	12.5917	0.9732	100	499

Ban. Palanka	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.0559	5.0858	0.9562	35	152
60	0.0557	5.0737	0.9565	49	209
120	0.0551	5.0256	0.9576	48	267
180	0.0550	5.0179	0.9580	93	469
365	0.0555	5.0105	0.9564	100	497

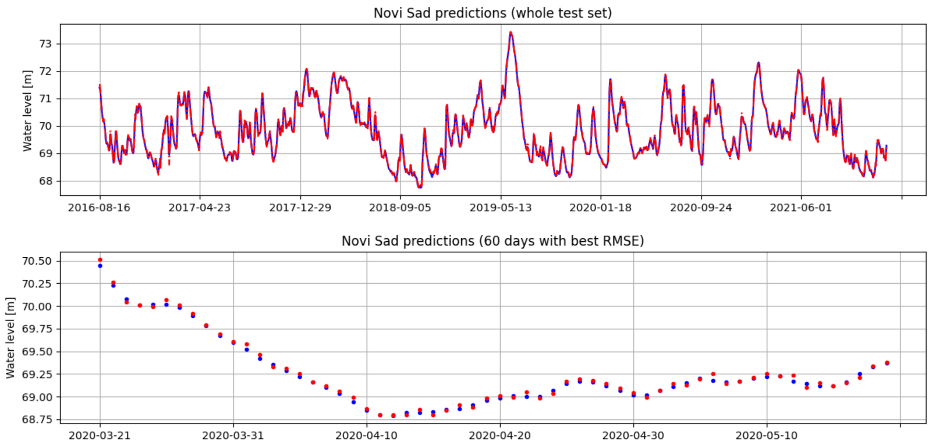
Prahovo	MAPE [%]	RMSE [cm]	R^2	Sigmoid neurons	RBF L2 neurons
30	0.4234	18.9374	0.9833	14	79
60	0.4210	18.9479	0.9836	29	163
120	0.8653	34.2724	0.9483	84	408
180	1.0182	40.1830	0.9301	87	363
365	2.0199	78.2243	0.7353	77	406


Bezdan predictions (whole test set)
Water level [m]
2016-08-16
2017-04-23
2017-12-29
2018-09-05
2019-05-13
2020-01-18
2020-09-24
2021-06-01
Bezdan predictions (60 days with best RMSE)
Water level [m]
2020-03-21
2020-03-31
2020-04-10
2020-04-20
2020-04-30
2020-05-10

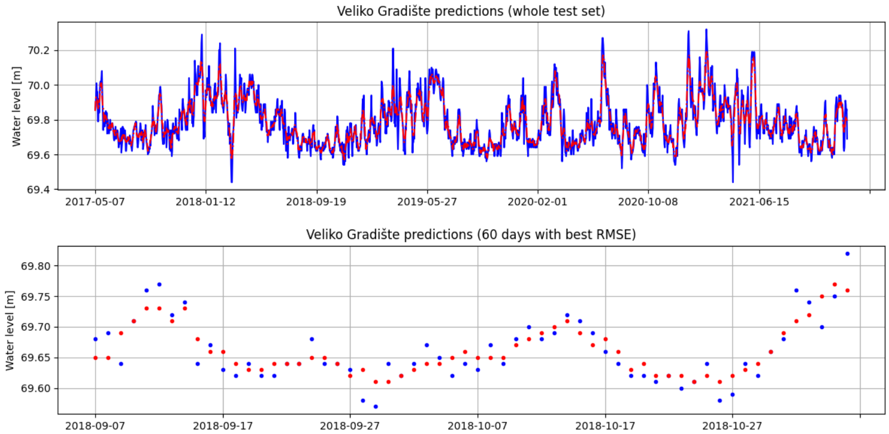
a)


Zemun predictions (whole test set)
Water level [m]
2018-02-15
2018-09-03
2019-03-22
2019-10-08
2020-04-25
2020-11-11
2021-05-30
2021-12-16
Zemun predictions (60 days with best RMSE)
Water level [m]
2018-03-21
2018-03-31
2018-04-10
2018-04-20
2018-04-30
2018-05-10

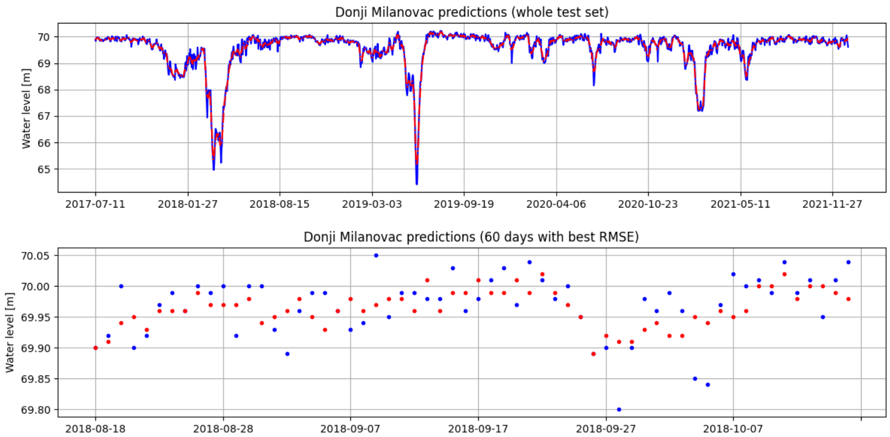
b)


Novi Sad predictions (whole test set)
Water level [m]
2016-08-16
2017-04-23
2017-12-29
2018-09-05
2019-05-13
2020-01-18
2020-09-24
2021-06-01
Novi Sad predictions (60 days with best RMSE)
Water level [m]
2020-03-21
2020-03-31
2020-04-10
2020-04-20
2020-04-30
2020-05-10

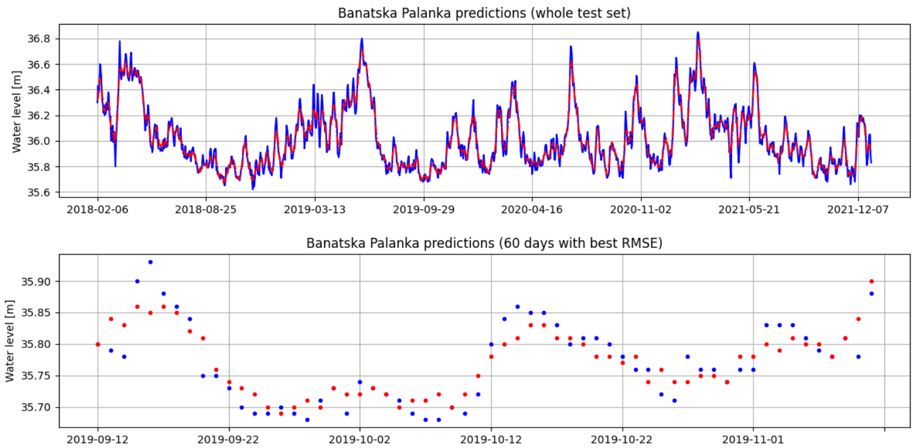
c)


Veliko Gradište predictions (whole test set)
Water level [m]
Veliko Gradište predictions (60 days with best RMSE)
Water level [m]

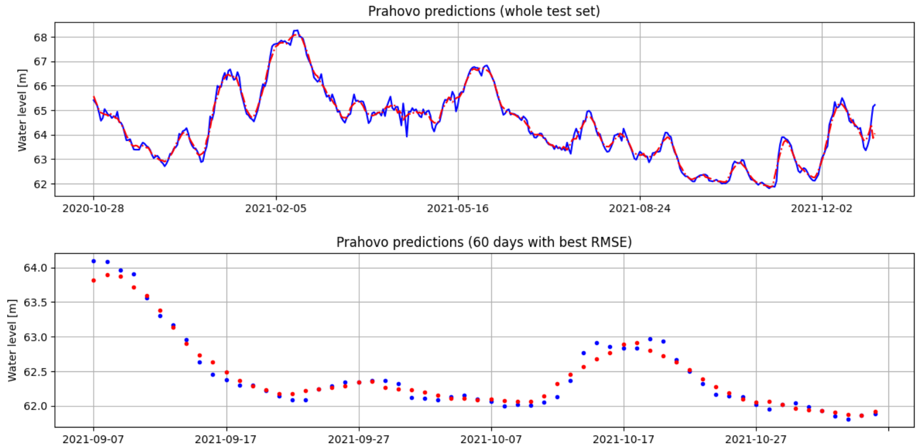
d)


Donji Milanovac predictions (whole test set)
Water level [m]
Donji Milanovac predictions (60 days with best RMSE)
Water level [m]

e)


Banatska Palanka predictions (whole test set)
Water level [m]
Banatska Palanka predictions (60 days with best RMSE)
Water level [m]

f)



g)

Fig. 5 Predictions for different models and best RMSE days – a) Bezdan model, b) Zemun model, c) Novi Sad, d) Veliko Gradište model, e) Donji Milanovac model, f) Banatska Palanka model, g) Prahovo model

7. CONCLUSION

This research examines the use of Extreme Learning Machine for predicting the daily water level of the Danube River. Numerous experiments with different ELM hidden neuron structures have been conducted across various datasets of different water level distributions. Models' accuracies were assessed in order to determine the optimal models for each dataset.

The ELM models demonstrated promising results in terms of prediction, considering their complexity, accuracy, and training time across most datasets. Most models showed desirable R^2 values (above 0.95), except for the Veliko Gradište station model, which had lower R^2 values due to higher data noise but low RMSE values. Regardless of the water level distribution or zero-elevation point, the models were able to accurately predict the water level.

For future studies, modifications of the models could be applied to further investigate the effect of different model parameters and properties on the models' predictive power. For instance, the number of output neurons could be increased in order to perform predictions for longer periods, such as one week or one month ahead. Since the models in this paper contained neurons of mixed activation functions, the effect of each activation function could be investigated by creating models with hidden layers with a single activation function and their performance can be compared with the models with mixed-activation function in hidden layers' neurons.

Acknowledgement: *This research was partially funded by the Ministry of Science, Technological Development and Innovations of the Republic of Serbia.*

REFERENCES

[1] J. Fabryka-Martin, and J. Merz, "The Study of Water and Water Problems: A Challenge for Today and Tomorrow", Universities Council on Water Resources, Hydrology, Carbondale, Ill., 1983.

[2] P. Dabral, and Z. M. Mharhoni. "Modelling and forecasting of rainfall time series using SARIMA", *Envir. Processes*, vol. 4, no. 2, pp. 399-419, 2017, doi.org/10.1007/s40710-017-0226-y.

[3] O. Bazrafshan, A. Salajegheh, J. Bazrafshan, M. Mahdavi, and A. F. Maraj. "Hydrological drought forecasting using ARIMA models (case study: Karkheh Basin)", *Ecopersia* vol.3, no. 3, pp. 1099-1117, 2015.

[4] W. Handoko, and A. N. Handayani. "Forecasting Solar Irradiation on Solar Tubes Using the LSTM Method and Exponential Smoothing", *J. Ilm. Tek. Elektro Komput. dan Inform*, vol. 9, no. 3, pp. 649-660, 2023, DOI: http://dx.doi.org/10.26555/jiteki.v9i3.26395.

[5] Y. Du, J. Wang, W. F. S. Pan, T. R. Xu, and C. Wang. "Adarnn: Adaptive learning and forecasting of time series", In *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 402-411. 2021.

[6] M. Bender, and S. Simonovic. "Time-series modeling for long-range stream-flow forecasting", *Journal of Water Resources Planning and Management*, vol. 120, no. 6, pp. 857-870, 1994, https://doi.org/10.1061/(ASCE)0733-9496(1994)120:6(857)

[7] D. Noakes, A. I. McLeod, and K. Hipel. "Forecasting monthly riverflow time series", *International Journal of Forecasting*, vol. 1, no. 2, pp. 179-190, 1985, https://doi.org/10.1016/0169-2070(85)90022-6.

[8] J. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso. "Deep learning for time series forecasting: a survey", *Big Data* vol. 9, no. 1, pp. 3-21. 2021, doi.org/10.1089/big.2020.0159.

[9] D. C. McKinney. "Modeling water resources management at the basin level: Review and future directions", International Water Management Institute, (1999), ISBN: 9789290903765.

[10] H. Maier, and G. C. Dandy. "Application of artificial neural networks to forecasting of surface water quality variables: issues, applications and challenges", *Artificial neural networks in hydrology (Book Section) - Springer*, pp. 287-309, 2000, doi.org/10.1007/978-94-015-9341-0_15

[11] R. Solgi, H. Loaiciga, and M. Kram. "Long short-term memory neural network (LSTM-NN) for aquifer level time series forecasting using in-situ piezometric observations", *Journal of Hydrology* vol. 601, October 2021, 126800, doi.org/10.1016/j.jhydrol.2021.126800.

[12] H. Han, C. Choi, J. Jung, and H. Kim. "Application of sequence to sequence learning based LSTM model (LSTM-s2s) for forecasting dam inflow", *Journal of Korea Water Resources Association* vol. 54, no. 3, pp. 157-166, 2021,

[13] K. Park, Y. Seong, Y. Jung, I. Youn, and C. K. Choi. "Development of Water Level Prediction Improvement Method Using Multivariate Time Series Data by GRU Model", *Water* vol. 15, no. 3 pp. 587, 2023, doi.org/10.3390/w15030587.

[14] K. Park, Y. Jung, Y. Seong, and S. Lee. "Development of deep learning models to improve the accuracy of water levels time series prediction through multivariate hydrological data", *Water* vol.14, no. 3, pp. 469, 2022, doi.org/10.3390/w14030469.

[15] N. Bezak et al., "A catalogue of the flood forecasting practices in the Danube River Basin", River Research and Applications, vol. 37, no. 7, pp. 909-918, 2021, doi.org/10.1002/rra.3826.

[16] T. Zabolotnia, L. Gorbachova, and B. Khrystiuk, "Estimation of the long-term cyclical fluctuations of snow-rain floods in the Danube basin within Ukraine", Meteorology Hydrology and Water Management. Research and Operational Applications, vol.7, no. 2, pp. 3-11, 2019.

[17] I. M. Santos et al., "Analysis of seasonal hindcasts for mean-term hydrological forecasting in the Upper Danube River Basin", Geophysical Research Abstracts, vol. 21, ISSN: 1029-7006, 2019.

[18] A. Z. Liptay, and B. Gauzer, "Operational River Ice and Water Temperature Forecasting on the Hungarian Danube Reach", FLOOD risk 2020 - 4th European Conference on Flood Risk Management, Budapest University of Technology and Economics, http://hdl.handle.net/10890/15156, 2021.

[19] D. Hussain et al., "A deep learning approach for hydrological time-series prediction: A case study of Gilgit river basin", Earth Science Informatics, vol. 13, pp. 915-927, 2020, doi.org/10.1007/s12145-020-00477-2.

[20] Z. Wang, and Y. Lou, "Hydrological time series forecast model based on wavelet de-noising and ARIMA-LSTM", In 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), pp. 1697-1701, IEEE, March, 2019.

[21] Z. M. Yaseen et al., "Non-tuned machine learning approach for hydrological time series forecasting", Neural Computing and Application, vol. 30, pp. 1479-1491, 2018.

[22] W. J. Niu et al., "Parallel computing and swarm intelligence based artificial intelligence model for multi-step-ahead hydrological time series prediction", Sustainable Cities and Society, vol. 66, 2021, doi.org/10.1016/j.scs.2020.1026.

[23] Z. K. Feng et al., "Evolutionary artificial intelligence model via cooperation search algorithm and extreme learning machine for multiple scales nonstationary hydrological time series prediction", Journal *of Hydrology*, vol. 595, 2021, doi.org/10.1016/j.jhydrol.2021.1260.

[24] J. Qin et al., "Simulating and Predicting of Hydrological Time Series Based on TensorFlow Deep Learning", Polish Journal of Environmental Studies, vol. 28, no. 2, 2019, DOI: 10.1524/pjoes/81557.

[25] M. Jeremić, M. Milić, J. Milojković, and M. Gocić, "Development of geo-da based android application for spatial and statistical analysis of Serbian long-term precipitation data", FACTA UNIVERSITATIS Series: Automatic Control and Robotics, vol. 21, no. 3, pp. 131-145, 2021, doi.org/10.2219/FUACR220623011J.

[26] M. D. Mijatović, Ž. Bjelajac, and I. Joksić, Overview of the ecological, economic and security capacities of the Danube River (in Serbian), Danubius, ISSN: 2217-4826, 2012.

[27] P. Pekarova et al., "Statistical analysis of hydrological regime of the Danube River at Ceatal Izmail Station", IOP Conference Series: Earth and Environmental Science. vol. 221, no. 1. IOP Publishing, 2019, DOI 10.1088/1755-1315/221/1/012035.

[28] T. Pulka et al., "A Near Real-Time Hydrological Information System for the Upper Danube Basin", Hydrology vol. 8, no. 4, 2021, doi.org/10.3390/hydrology8040144.

[29] ICPDR. Danube Basin: Facts & Figures. International Commission for the Protection of the Danube River. Available online: https://www.icpdr.org/flowpaper/viewer/default/files/nodes/documents/icpdr_facts_figures.pdf (accessed on 1 December 2020).

[30] P. Pekárová et al., "Identification of long-term high-flow regime changes in selected stations along the Danube River", Journal of Hydrology and Hydromechanics, vol. 64, no. 4, pp. 393-403, 2016, DOI: 10.1515/johh-2016-0045.

[31] J. Wesemann, H. Holzmann, K. Schulz, and M. Herrnegger, "Behandlung künstlicher Speicher und Überleitungen in der alpinen Niederschlags-Abfluss-Vorhersage", Osterr. Wasser Abfallwirtsch. vol.70, no. 11, pp. 485-496, 2018, DOI: 10.1007/s00506-018-0501-9.

[32] D. Bănăduc et al., "The Danube Delta: The Achilles Heel of Danube River–Danube Delta–Black Sea Region Fish Diversity under a Black Sea Impact Scenario Due to Sea Level Rise—A Prospective Review", Fishes, vol. 8, no. 7, 2023, doi.org/10.3390/fishes8070355.

[33] A. M. Petrović, I. Novković, and S. Kostadinov, "Hydrological analysis of the September 2014 torrential floods of the Danube tributaries in the Eastern Serbia", Natural Hazards, vol. 108, pp. 1373-1387, 2021.

[34] K. P. Schneider et al., "A Distributed Power System Control Architecture for Improved Distribution System Resiliency", IEEE Access, vol. 7, pp. 9957-9970, DOI: 10.1109/ACCESS.2019.2891, 2019.

[35] A. Bidram, F. L. Lewis and A. Davoudi, "Distributed Control Systems for Small-Scale Power Networks: Using Multiagent Cooperative Control Theory," IEEE Control Systems Magazine, vol. 34, no. 6, pp. 56-77, Dec. 2014, doi: 10.1109/MCS.2014.2350.

[36] Flood recovery and prevention, Studie, https://neighbourhood-enlargement.ec.europa.eu/system/files/2016-12/ipa2014_037788.06_rs_floods_recovery_and_prevention.pdf.

[37] G. B. Huang, Q. Y. Zhu and C. K. Siew, "Extreme Learning Machine: Theory and Applications", Neurocomputing, vol. 70, no. 1, pp. 489-501, 2006, doi.org/10.1016/j.neucom.2005.12.126.

[38] C. W. Deng, G. B. Huang, J. Xu, and J. X. Tang, "Extreme learning machines: new trends and applications" Science China, Information Sciences, vol. 58, no. 2, pp. 020301–020301, 2015, doi.org/10.1007/s11432-014-5269-3.

[39] G. B. Huang, D. H. Wang, and Y. Lan, "Extreme Learning Machines: A Survey", International Journal of Machine Learning and Cybernetics, vol. 2, no. 2, pp. 107-122, 2011, doi.org/10.1007/s13042-011-0019-y.

[40] Aranildo R. Lima, Alex J. Cannon, William W. Hsieh, "Nonlinear regression in environmental sciences using extreme learning machines: A comparative evaluation", Environmental Modelling & Software, vol. 73, pp. 175-188, 2015, doi.org/10.1016/j.envsoft.2015.08.002

[41] S. Ding, H. Zhao, Y. Zhang, X. Xu and R, Nie, "Extrem.e learning machine: algorithm, theory and applications", Artificial Intelligence Review, vol. 44, no. 1, pp. 103–115, 2015, doi.org/10.1007/s10462-013-9405-z.

[42] M. van Heeswijk, Y. Miche, E. Oja, A. Lendasse, "GPU-accelerated and parallelized ELM ensembles for large-scale regression", Neurocomputing, vol. 74, no. 16, pp. 2430-2437, 2011, doi.org/10.1016/j.neucom.2010.11.034.

[43] J. C. A. Barata, M. S. Hussein, "The Moore-Penrose Pseudoinverse. A Tutorial Review of the Theory", Brazilian Journal of Physics, vol. 42, pp. 146-165, 2011, doi.org/10.4855/arXiv.1110.6882

[44] Annual reports for hydrological data of Danube River 1991-2021 (in Serbian), available online at: https://www.hidmet.gov.rs/ciril/hidrologija, Accessed on 10th of December 2023.

[45] "Operational jobs in general working program of surface water hydrological stations network", Republic Hydrometeorological Service of Serbia, available online at: https://www.hidmet.gov.rs/latin/hidrologija/delatnost_mreza.php, Accessed on 10th of January 2024.

[46] M. Andrejević Stošović, N. Radivojević, I. Jovanović and A. Petrušić, "Artificial Neural Networks Application to Prediction of Electricity Consumption". Facta Universitatis, Series: Automatic Control and Robotics, 20(1), 033-042. Doi: HTTPs://doi.org/10.2219/FUACR201231003A.
[47] Y. Miche et al., "OP-ELM: Optimally Pruned Extreme Learning Machine", in IEEE Transactions on Neural Networks, vol. 21, no. 1, pp. 158-162, Jan. 2010, DOI: 10.1109/TNN.2009.2036.
[48] T. Similä and J. Tikka, "Multiresponse sparse regression with application to multidimensional scaling", in Proc. Int. Conf. Artif. Neural Netw., pp. 97–102, 2005 DOI:10.1007/11550907_16.
[49] Ch. S. K. Dash, A. K. Behera, S. Dehuri and S. B. Cho, (2016). "Radial basis function neural networks: A topical state-of-the-art survey", Open Computer Science, vol. 6, no. 1, 2016, DOI 10.1515/comp-2016-0005.