# ARTIFICIAL NEURAL NETWORK-BASED DAILY RAINFALL-RUNOFF MODELLING CASE STUDY OF THE TESSA WATERSHED, SEMI-ARID REGION, TUNISIA

KOTTI, M. L.[1,2*] – HERMASSI, T.[1]

[1]*National Research Institute of Rural Engineering, Water and Forests (INRGREF), Hedi El Karray street El Menzah IV- P.O. Box N°10-2080 Ariana, Tunisia (phone: +216-71-719-630; fax: +216-71-717-951)*

[2]*University of Carthage, National Agronomic Institute of Tunis, 43 Avenue Charles Nicolle 1082-Tunis Mahrajen, Tunisia*

[*]*Corresponding author*
*e-mail: mohamed.l.kotti@gmail.com*

(Received 1st Feb 2024; accepted 16th May 2024)

**Abstract.** In semi-arid areas, accurate streamflow forecasting is critical for planning and managing water resources. Streamflow is very complex and non-linear in nature and modelling tools may fail to represent this complexity while maintaining the reliability of the data set. This current study attempts to increase modelling accuracy and reduce uncertainties using Artificial Neural Network (ANN). ANN could efficiently manage non-linearity in big and complicated datasets to simulate the rainfall-runoff process using data as well as daily rainfall and hydrometric data. The proposed method is used to improve the accuracy of daily streamflow forecasts in the Tessa watershed in Tunisia. The hydrological modelling process parameters used are daily rainfall from 12 rainfall stations, evaporation, and discharge from one hydrometric station. Daily rainfall, evaporation and discharge data at the current and previous time steps were used as input parameters, with the current discharge as the output parameter. Performance evaluation study was carried out using Nash-Sutcliff efficiency coefficient (NSE), correlation coefficient (R), Mean average error (MAE) and Root Mean Square Error (RMSE). The results reveal that an ANN model with an input combination of daily rainfall, evaporation, and lag one streamflow and evaporation (M7, M7-H1) outperforms other models. In fact, those models can forecast daily streamflow with greater accuracy and captures effectively the non-linear and complicated hydrological time series structure for a semi-arid region.
**Keywords:** *neural network, watershed, modeling, accuracy, rainfall, Tunisia*

## Introduction

It is difficult to determine the process by which rainfall produces runoff since there are so many pertinent elements that fluctuate over space and time. Accurate evaluation of this process is necessary for the rational management of the various uses of water, including supply, irrigation, the production of electricity, and the forecasting of extreme flood occurrences and dry spells. The examination of this process is typically carried out using mathematical models referred to as rainfall-runoff models.

Because the interrelationships between the numerous subprocesses (engaged in the hydrologic cycle) are so complicated and the rainfall-runoff process is so highly nonlinear, conceptual models are not always appropriate for modeling it (Zhang et al., 2000). Moreover, various conceptual rainfall-runoff models are computationally intensive and require a lot of data for both calibration and validation process, which prevents them from gaining much traction (Lu et al., 2012).

Rainfall-runoff models can be categorized into two main categories: empirical and conceptual models. Based on the physical principles that regulate each of these activities,


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

the conceptual models provide a mathematical description of the processes in the hydrological cycle. Nevertheless, the overall favorable results, some parts of those conceptual models are challenging. Calibration process is complex and usually depends on field studies using limited amounts of data. Further challenges arise from the nonlinear nature of those processes and the use of basin averages for pertinent parameters. These characters frequently make the execution of the conceptual model challenging and expensive.

Conceptual models can be substituted by empirical models. The main feature of these specific models lies in establishing a dependable connection between input and output variables, disregarding the physical principles governing the generation of runoff from rainfall. These models are simple to use and described as less expensive. Multivariable equations with least-squares-estimated parameters and artificial neural networks (ANNs) are two examples of these models.

In such cases, the use of an Artificial Intelligence (AI) method might be considered as a viable alternative technique for modeling and enhancing the predictive precision of intricate hydrologic systems using extensive datasets (Wagena et al., 2020). Hydrology makes extensive use of AI-based data driven models, particularly for rainfall-runoff prediction (Poonia and Tiwari, 2020), sediment transport and concentration (Nagy et al., 2002; Ebtehaj et al., 2020), river flow forecasting (Adeyemo et al., 2018), and groundwater forecasting (Ghazi et al., 2021). ANN are widely used data-driven model for hydrological forecasts due to their adaptability and ability for real-time analysis. They are more effective than other methods such as regression models and time-series models.

The structure known as an artificial neural network (ANN) is composed of mathematically connected nodes or neurons that determine a function. Weights and biases are the names given to the coefficients and captures of the input variables for this function. There are other ANN types, but the most prevalent is the multilayer perceptron (MLP), which divides the neurons into layers, usually three (de O. Galvão, 1999). Here, a multilayer perceptron ANN is identified using the ANN terminology (MLP). With its nonlinear properties, a three-layer ANN is capable of handling any function, according to Fernandes et al. (1996) and de O. Galvão (1999).

ANNs have been employed in the discipline of water management to resolve a variety of issues, including inflow predicting and managing reservoirs (Jain et al., 1999), reservoir operation simulation and optimization (Neelakantan et al., 2000). In hydrology, forecasting streamflow or rainfall is one of ANNs' main applications. Many works on the subject include those by Cigizoglu (2003a,b), Dakhlaoui et al. (2012) and Warwade et al. (2018). Other studies used the ANN for the simulation of river quality parameters such as those by May et al. (2008) and Wu et al. (2014), also for the estimation of the sediment transport by Alp and Cigizoglu (2007). An ANN was employed by Minns and Hall (1996) to simulate the peak discharge and the flood hydrograph. The results demonstrated that the ANN’s performance was higher for the peak discharge simulation than the flood hydrograph simulation (as a result of network optimization based on peak discharge). The study shows also that the study's objective influences the target function selection and has a key role in the error rate of the calculated hydrograph (Jain and Prasad Indurthy, 2003; Rezaeianzadeh et al., 2013). More recently, Sharma et al. (2021) showed how to predict the monthly streamflow in India for the Sot River basin, for eight years (2009 to 2016) using artificial neural networks (ANNs) and backpropagation methods. They used several input-output parameter combinations, and the results were considerably enhanced by including the lag-one flow as one of the input factors.


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

The semi-arid region of Tunisia, where this research is being conducted, is distinguished by erratic precipitation patterns that significantly affect the hydrological cycle and the availability of water resources. Furthermore, rainfall is often unpredictable in such places, causing significant fluctuations in streamflow. Additionally, considering the very complicated and non-linear streamflows, calibrating hydrological parameters for semi-arid locations is particularly difficult. This study proposed an artificial neural network-modelling framework that takes two elements into account: (1) data calibration and (2) validation. A huge number of possible combinations of input parameters are used in the model creation process. Finding the optimal arrangement for the input data is the main goal of this study, the optimal architecture of an Artificial Neural Network for streamflow forecasting and also to improve accuracy and reduce uncertainty for the semi-arid region of TESSA. The approach that was used in this investigation is detailed below.

## Materials and Methods

### *Study area and data description*

At the Sidi Medien stream gauging station, the case study was created specifically for the Tessa River basin. The Tessa watershed is located between latitude (3960000-4040000) and longitude (480000-520000) with the Carthage-UTM zone 32 projection (*Fig. 1*). The choice of The Tessa River was based on: (i) it is one of the Medjerda's river right bank tributaries and its significant contribution to the flooding of the middle valley of the Mejerda; (ii) the density of rainfall stations: the Tessa basin has fourteen rainfall gauging stations and one stream gauging station located approximatively in the middle of the watershed; and (iii) for data series' quality and extent: the daily data for the period September 1995-August 2003 were used for training, validation and test process. The effect of the database's length for each process has been studied and developed in the rest of this paper. The pertinent rainfall-runoff data statistics are displayed in *Table 1*.

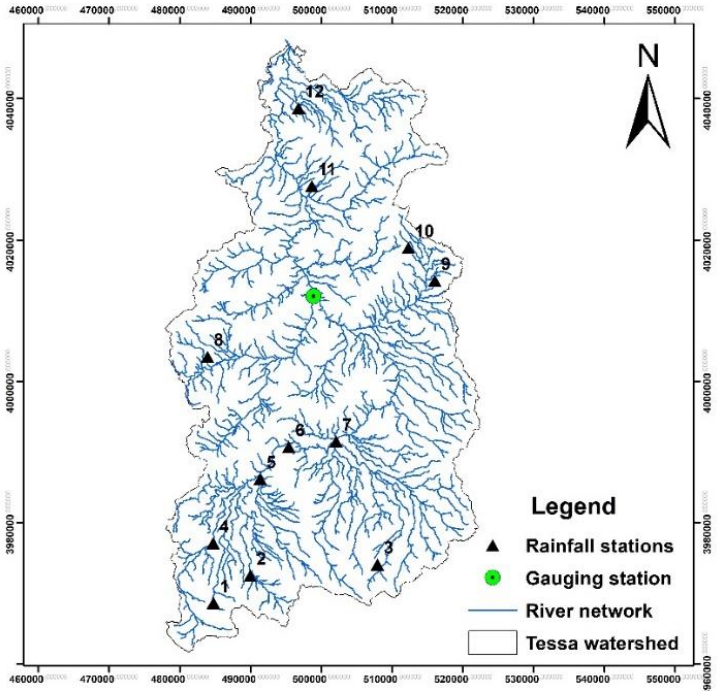


***Figure 1.*** *Tessa watershed location*


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

***Table 1.*** *Statistic of daily rainfall-runoff data for the period between September 1995 and August 2003 period*

| | Rainfall (mm) | Flow(mm) |
|---|---|---|
| Mean | 1.22 | 0.16 |
| St.dev | 4.54 | 1.225 |

For this study, historical streamflow daily rainfall and meteorological database for the period of 8 years (1995-2003) are obtained from the General Directorate of Water Resources (*Table 2*). This period was chosen to study the contribution of the Tessa watershed to the flood event that affected the Medjerda basin in 2003. The rainfall data presented a maximum rainfall equal to 86.5 mm recorded for the Hammam Byadha station, and a minimum of 0 mm recorded for all the rainfall stations. the Sidi Medien hydrometric station has a maximum recorded flow equal to 29.31 mm and a minimum of 0 mm with an average of 0.17 mm for the period between 1995 and 2003. Initially, the flow data recorded at the Sidi Medien hydrometric station are expressed in (m3/s), hence the shift towards determining specific flows in (mm) to ensure consistency across all the units in the study.

***Table 2.*** *Essential information concerning the rainfall measuring stations and the daily rainfall statistics analysis (1995-2003)*

| N° | Station Name | Elevation (m) | Lat. (UTM m) | Lon. (UTM m) | Max | Mean |
|---|---|---|---|---|---|---|
| 1 | Ain Zeligua | 853 | 484756 | 3968624 | 51 | 1.10 |
| 2 | Ksour Ecole | 720 | 489951 | 3972498 | 59 | 1.21 |
| 3 | ELLES Ecole Sers | 695 | 507923 | 3974068 | 52 | 0.86 |
| 4 | Dehmani | 622 | 484695 | 3977097 | 54 | 1.19 |
| 5 | Zouarine Gare | 571 | 491343 | 3986176 | 70 | 1.18 |
| 6 | Fath Tessa | 532 | 495400 | 3990702 | 58.2 | 1.14 |
| 7 | Sers Delegation | 501 | 502079 | 3991472 | 52 | 1.28 |
| 8 | Zaafrane UCP | 530 | 483917 | 4003471 | 82 | 1.47 |
| 9 | Ain Tabia | 416 | 516019 | 4014224 | 72 | 1.26 |
| 10 | Krib Ferme Cossem | 447 | 512295 | 4018871 | 66.2 | 1.51 |
| 11 | Hammam Bayadha | 240 | 498633 | 4027613 | 86.5 | 1.56 |
| 12 | SK El Khemis | 146 | 496769 | 4038644 | 60 | 1.30 |

Daily precipitation, flow discharge, and evapotranspiration were selected as input data for the study case. Consistency checks were performed on all the data:

- The invalid data were removed in the case of precipitation. Due to their numerous gaps, two out of the fourteen stations were not used in this study. Using twelve stations and the Thiessen method, the mean precipitation was calculated, with data spanning 2920 days between September 1995 and August 2003;
- Based on the Penman method, the evapotranspiration was computed, showing respectively the minimum, maximum and average values of 1.012 mm, 4.7 mm and 2.488 mm.
- The data collected at the Tessa stream gauging station are used to calculate the mean daily discharge (*Fig. 1*), with respective minimum, maximum and average values of around 0.00 mm, 29.31 mm and 0.17 mm. For every rainfall event, the basin response was assessed qualitatively (visually figure representing rainfall/runoff) and statistically (correlation rainfall/runoff) to analyze the overall behavior.


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

### *Artificial neural network modelling*

The proposed approach is fully explained in this section, along with a brief introduction to ANN modelling, dataset preparation, ANN design training settings, and performance metrics for assessing the effectiveness of the created models.

ANNs are computational models imitating the inner workings of the human nervous system. Made up of a significant number of processing elements (PEs), they are also known as artificial neurons that are both straightforward and densely coupled (Chen et al., 2013). For ANN, an artificial neuron serves two main purposes. As seen in *Fig. 2*, it first employs an activation function to the neuron's net input, which is the weighted total of all of its inputs.

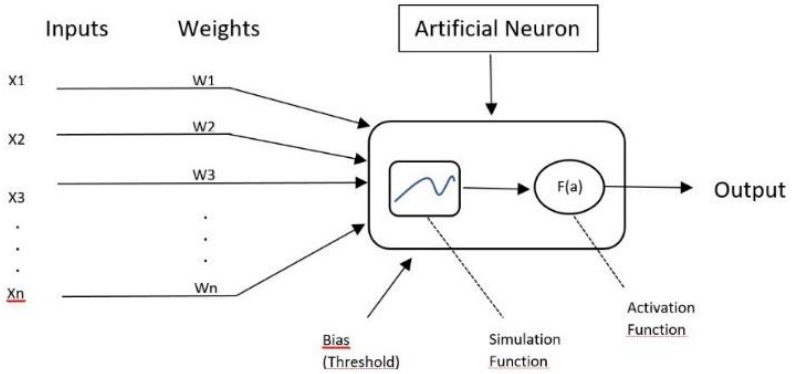


***Figure 2.*** *A schematic illustration of an artificial neuron's mathematical model*

The use of artificial neural networks (ANNs) for modeling complex relationships has been widespread since a few decades ago. These scenarios include speech recognition, currency price prediction, feature extraction, and other situations where it is extremely challenging to establish the mathematical relationship for any physical phenomenon between input and output factors (Khan et al., 2016). For this study, a multilayer perception neural network, known as MLP, was created to model rainfall-runoff, applying a feed-forward back propagation method. The specific type of ANN was selected due to its better compared to other ANN models. The frequently employed artificial neural network (ANN) technique for modeling hydrological activities is the Multilayer Perceptron (MLP). An MLP's architecture is made up of an input layer that holds all of the model's input data, one or more hidden layer(s) and an output layer. *Fig. 3* shows a MLP architecture with a single hidden layer.

The input data is sequentially passed through the ANN in a unidirectional manner, moving layer by layer in the forward direction. This is known as the feed-forward data flow. The corresponding connection's weights are multiplied by the inputs in the input layer. In the intermediate layers, each neuron computes a linear combination of the input variables. The combination in question activates the transfer function, which generates an output. The transfer function answers are the inputs to the following layer. The output layer's input is a linear combination of the middle layer's outputs. The ANN response is the output of the output layer (Machado et al., 2011). The mathematical components of a three-layer ANN's output are represented by *equation (1)*.

$$y_k = \varphi(\sum_{j=1}^{q} w_{kj}\varphi(\sum_{i=1}^{p}(w_{ji}x_i + b_j) + b_k) \quad \text{(Eq.1)}$$


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

where $y$ is the ANN output and $i$, $j$ and $k$ are neurons of the input, middle and output layers, respectively, $\varphi$ is the transfer function, $p$ the number of neurons in the input layer, $q$ the number of neurons in the middle layer, $w$ are the weights between the connections, $x$ are the input elements and $b$ the biases.

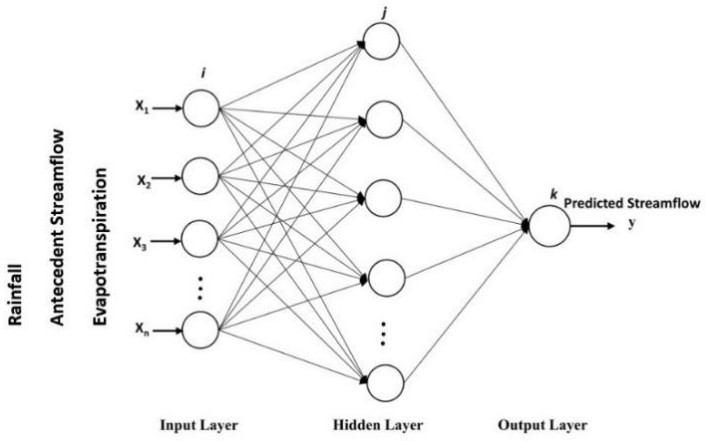


***Figure 3.*** *Rainfall-Runoff modelling using MLP architecture with one hidden layer*

The backpropagation algorithm is a widely recognized ANN training algorithm. It employs supervised learning, indicating that it learns from labeled training data under the guidance of a supervisor. At the start of running ANN model trains with random weights are applied to all connections, and the goal is to alter these randomly generated weights for connections so that the gap between the expected and observed values, is as small as possible. A backpropagation learning algorithm is used to accomplish that.

A forward pass and a backward pass are the two stages of the backpropagation algorithm. During the forward pass, samples of input are fed into the model and transferred from the input to the output layer. Following that, a loss function computes the difference between the model's estimated and observed outputs to estimate an error. During the backward pass, the calculated output layer's error is propagated backward from the output layer to the input layer. Error's backward propagation is similar to learning from mistakes. To minimize the loss function, the supervisor changes the model weights when the network produces an inaccurate output. Attending a loss function's value below a specified threshold value, the training process is halted.

The backpropagation technique is employed to train this model, signifying that it acquires from training data and undergoes backward error propagation to rectify its mistakes. For the backpropagation training process, the Levenberg-Marquardt (LM) algorithm was implemented employing the 'trainlm' function in MATLAB (R2015a). Moreover, a logistic sigmoid function has been applied in both the hidden and output layers. The input and target files were organized in CSV file format.

In order to study the effect of the length of the time series data set allocated for the training phase, three scenarios were initially developed. The first scenario consisted of dividing the database into 10% for training and 90% for the validation and testing phase. The second scenario divided the database into 50% for the training phase and 50% for validation and testing. Finally, the third scenario consisted of allocating 70% of the database to the training process and 30% to validation and testing (70% training, 15% validation and 15% testing). To execute the script, this division has been made by


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

chronological block. The model's training was done by setting the learning rate, the maximum number of epochs and the validation check number to 0.001, 1000, and 20, respectively. To explore the most effective operational configuration, 36 artificial neural networks (ANNs) were generated, varying in the input's number (five combinations) and the number of neurons in the intermediate layers (four options).

The runoff is maintained as the output of the model. Each pairing of input and output was designated as a model, and *Table 3* illustrates the input and output for each of these models. Runoff (Q), Precipitation (P), and evapotranspiration (EVT) serve as the input variables, all represented as daily values. The evapotranspiration and the precipitation were both used for the current preceding time steps, while the input for the runoff was established at the previous time step which could be considered as indicator of the basin's moisture content. For each model, the middle layer incorporated three, five, eight, and ten neurons. The evaluation of ANN sensitivity was made by changing the number of the middle layer's neuron, hidden layer size and the number of inputs.

***Table 3.*** *ANN models proposed*

| Model | Inputs | Outputs |
|---|---|---|
| M1 | P(t) | Q(t) |
| M2 | P(t), EVT(t) | Q(t) |
| M3 | P(t), Q(t-1) | Q(t) |
| M4 | P(t), EVT(t), Q(t-1) | Q(t) |
| M5 | P(t-1), P(t), EVT(t), Q(t-1) | Q(t) |
| M6 | P(t-2), P(t-1), P(t), EVT(t), Q(t-1) | Q(t) |
| M7 | P(t), EVT(t), EVT(t-1), Q(t-1) | Q(t) |
| M8 | P(t-1), P(t), EVT(t), EVT(t-1), Q(t-1) | Q(t) |
| M9 | P(t-1), EVT(t), EVT(t-1), Q(t-1) | Q(t) |

### *Performance evaluation of ANN model*

For this study, the performance of the MLP model in the course of testing is evaluated using three statistical criteria: Nash-Sutcliffe efficiency (NSE), the correlation coefficient (R), the mean absolute error (MAE), and the root-mean-squared error (RMSE) which are defined as:

$$NSE = 1 - \frac{\sum_{i=1}^{n}(d_i - y_i)^2}{\sum_{i=1}^{n}(d_i - \bar{d})^2} \quad \text{(Eq.2)}$$

$$R = \frac{\sum_{i=1}^{n}(d_i - \bar{d})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(d_i - \bar{d})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad \text{(Eq.3)}$$

$$MAE = \frac{\sum_{i=1}^{n}|y_i - d_i|}{n} \quad \text{(Eq.4)}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(d_i - y_i)^2}{n}} \quad \text{(Eq.5)}$$

where i is the time variable for streamflow; n is the total number of data samples, and d,$\bar{d}$, y and $\bar{y}$ denote observed streamflow, mean observed streamflow; predicted streamflow, and mean predicted streamflow at ith time interval, respectively.


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

## Results

The training dataset is employed to train all of the models in this study. To avoid model overfitting, the validation dataset is employed alongside the training dataset throughout the model-training phase. The test dataset is utilized to assess the performance of the MLP models. *Table 4* displays information about the MLP models, outlining their architecture and performance metrics. Each MLP model architecture is defined by the number of neurons in the input, hidden, and output layers. Neurones input's layer number corresponds to the count of input patterns introduced to the MLP neural network model, and this varies for each of the twenty MLP models. In contrast, the output layer neuron generates the projected streamflow as the MLP model's output. M3-1 has a 2-3-1 architecture, which means it has 1 input neuron, 3 hidden layer neurons, and 1 output neuron. The aggregated rainfall data from all climatic stations across the research period is represented by the single input neuron (refer to *Table 3*). Similarly, the 4-8-1 M7-3 design denotes the existence of 5, 8 and 1 neurons in the input, hidden, and output layers respectively. M7 is built by combining evaporation, streamflow data from a gauging station for the previous day, evaporation data, and the current day's rainfall data, as presented in *Table 3*.

***Table 4.*** *Influence of the number of input parameters and the ANN structure*

| Model | Structure | NSE | R | RMSE | MAE |
|---|---|---|---|---|---|
| M3 | 2-3-1 | -0.9727 | 0.66019 | 0.9462 | 0.2485 |
| | 2-5-1 | 0.2083 | 0.7486 | 0.5995 | 0.2459 |
| | 2-8-1 | -0.2065 | 0.77389 | 0.74 | 0.3486 |
| | 2-10-1 | -0.4048 | 0.68601 | 0.7985 | 0.2485 |
| M4 | 3-3-1 | 0.3338 | 0.75918 | 0.5499 | 0.6358 |
| | **3-5-1** | **0.4508** | **0.7542** | **0.4993** | **0.1886** |
| | 3-8-1 | 0.0566 | 0.77162 | 0.6544 | 0.6325 |
| | 3-10-1 | 0.3621 | 0.7518 | 0.5381 | 0.2168 |
| M5 | 4-3-1 | -31.9213 | 0.49293 | 3.8656 | 0.5482 |
| | 4-5-1 | -142.1611 | 0.38563 | 8.5963 | 0.7256 |
| | 4-8-1 | -5.128 | 0.71226 | 1.6678 | 0.2862 |
| | 4-10-1 | -83.406 | 0.39651 | 6.1896 | 0.6632 |
| M6 | 5-3-1 | -26.2394 | 0.55675 | 3.5162 | 0.4479 |
| | 5-5-1 | -50.5381 | 0.50872 | 4.8366 | 0.4765 |
| | 5-8-1 | -40.1394 | 0.54159 | 4.3112 | 0.4259 |
| | 5-10-1 | -1.141 | 0.76977 | 0.9858 | 0.263 |
| M7 | **4-3-1** | **0.4786** | **0.76031** | **0.4865** | **0.1903** |
| | 4-5-1 | 0.286 | 0.77373 | 0.5693 | 0.2053 |
| | 4-8-1 | 0.3487 | 0.77223 | 0.5437 | 0.2085 |
| | 4-10-1 | -1.0435 | 0.74054 | 0.9631 | 0.2338 |
| M8 | 5-3-1 | -1.7269 | 0.78012 | 1.1125 | 0.286 |
| | 5-5-1 | -23.9215 | 0.58042 | 3.3633 | 0.4405 |
| | 5-8-1 | -3.9356 | 0.76591 | 1.4967 | 0.2883 |
| | 5-10-1 | -19.6645 | 0.62027 | 3.0626 | 0.4943 |
| M9 | 4-3-1 | 0.0099 | 0.42272 | 0.6704 | 0.2396 |
| | 4-5-1 | -33.2386 | 0.50974 | 3.9421 | 0.4689 |
| | 4-8-1 | -71.3407 | 0.42184 | 5.7301 | 0.6398 |
| | 4-10-1 | -90.3742 | 0.43861 | 6.44 | 0.6699 |


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

The initial step of the research was to investigate the relationship and also the effect of the input parameter's number, their optimal combination, and the length of the data set provided for the model training phase on the projected results. The greatest results were achieved using a training data set that was 70% of the whole data collection. According to *Table 4*, Models M4 and M7 produced the best results with three and four input data sets, respectively: M4: P(t), EVT(t), Q(t-1); M7: P(t), EVT(t), EVT(t-1), Q(t-1).

Based on the data as presented in *Table 4*, it is evident that the M7 model has the highest NSE value, which is 0.4786. The M7 model was adopted for the remainder of this study to explore the influence of the number of validation check numbers, the hidden layer neurons number as well as the effect of the size of the hidden layer.

As mentioned, the initial value of the validation check number was initially set at 20, and the effect of varying this parameter was studied by varying it in the range [5, 10, 40, 60, and 80]. The results showed that increasing the validation check value from 5 to 20.

led to an improvement in the NSE values, but above the value of 20 and for values of 40, 60, 80 this increase led to a spectacular decrease in the NSE value, as shown in *Figure 4*.

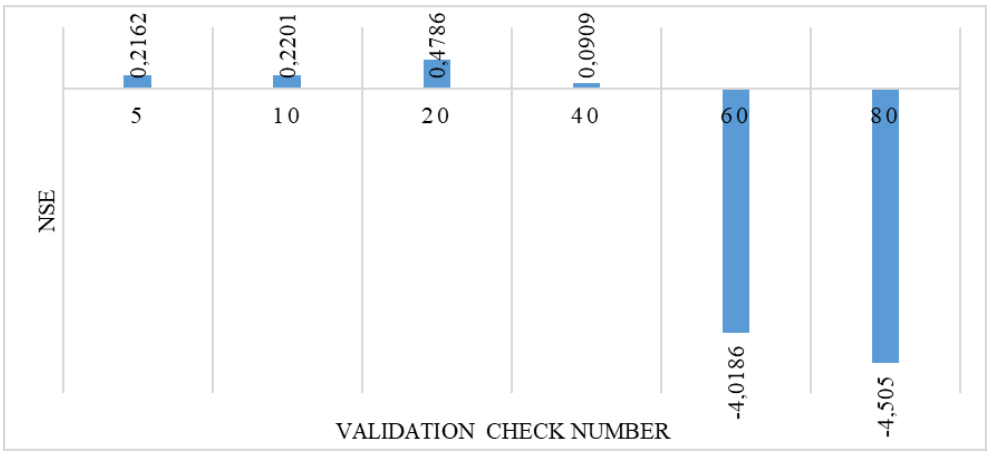


***Figure 4.*** *Validation check number effect on the result for M7 ANN model*

The second phase of this study involved examining the impact of incrementally increasing the number of neurons from 1 to 10, with a step size of 1. *Figure 5* depicts the NSE as a function of the number of intermediate neurons, which was employed as the primary selection criterion in this study. The M7 variant with the 4-3-1 configuration consistently gets the best NSE. Following this observation, we were interested in investigating the effect of increasing the size of the hidden layer for a second and third degree while maintaining the same input and output data. We created an intermediate level between the baseline structure and the desired values for the second degree, and we adjusted the number of neurons in this level from 1 to 10 in steps of 1 neuron each time. *Table 5* shows the results of this change, which demonstrate an improvement in the results for the M7-H2 structure (4-3-1-1) with an NSE of 0.52.

The M7-H1 model was used as the basic model for the third degree, and the same process was followed. The results revealed that extending the level to a third degree had a detrimental effect on the results, with a maximum of 0.3668 (*Table 6*) for the M7-H1-1 (4-3-1-1-1) structure.


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

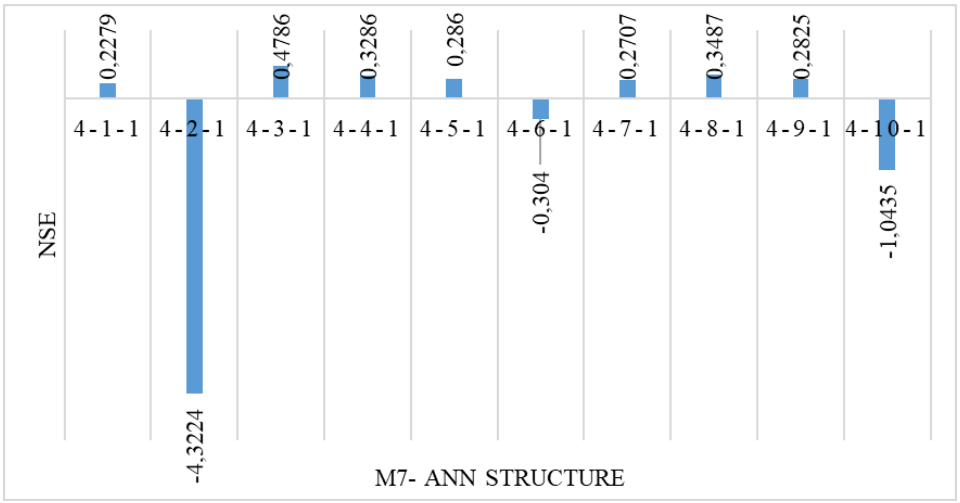


***Figure 5.*** *Effect of the number of hidden neuron on the results*

***Table 5.*** *Effect of hidden size level (Second level)*

| Model | Hidden size 2-level | Structure | NSE |
|---|---|---|---|
| M7 | H0 (Initial Structure) | 4-3-1 | 0.4786 |
| | H1=[3,1] | 4-3-1-1 | **0.52** |
| | H2=[3,2] | 4-3-2-1 | 0.0256 |
| | H3=[3,3] | 4-3-3-1 | -3.5763 |
| | H4=[3,4] | 4-3-4-1 | 0.4049 |
| | H5=[3,5] | 4-3-5-1 | 0.3487 |
| | H6=[3,6] | 4-3-6-1 | 0.4258 |
| | H7=[3,7] | 4-3-7-1 | 0.2479 |
| | H8=[3,8] | 4-3-8-1 | 0.2668 |
| | H9=[3,9] | 4-3-9-1 | 0.3841 |
| | H10=[3,10] | 4-3-10-1 | 0.2725 |

***Table 6.*** *Effect of third-degree level size on the results*

| Model | Hidden size 3-level | Structure | NSE |
|---|---|---|---|
| M7 | H1-1= [3,1,1] | 4-3-1-1-1 | **0.3668** |
| | H1-2= [3,1,2] | 4-3-1-2-1 | 0.0993 |
| | H1-3= [3,1,3] | 4-3-1-3-1 | -0.0191 |
| | H1-4= [3,1,4] | 4-3-1-4-1 | -0.2864 |
| | H1-5= [3,1,5] | 4-3-1-5-1 | -1.3096 |
| | H1-6= [3,1,6] | 4-3-1-6-1 | -0.1336 |
| | H1-7= [3,1,7] | 4-3-1-7-1 | -5.8764 |
| | H1-8= [3,1,8] | 4-3-1-8-1 | 0.1056 |
| | H1-9= [3,1,9] | 4-3-1-9-1 | -0.0168 |
| | H1-10= [3,1,10] | 4-3-1-10-1 | -3.1254 |

*Figures 6* and *7* provide a scatter plot comparing the hydrological modelled streamflow to the observed streamflow of ANN M7 an M7-H1 models. The hydrographs in *Figures 6* and *7* show that simulated streamflows were underestimated in comparison to observed streamflows. This demonstrated that the M7 and M7-H1 ANN models performed well when modeling low flows. Additionally, except for elevated flows in both the calibration


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

and validation periods, the simulated streamflow closely resembles the observed streamflow. Sharma (2021) made comparable assumptions. In terms of peak flows, the highest values recorded are 8.04 and 6.66 mm. The M7 and M7-H1 models duplicate the peak flows on the same date but with substantially lower values of 4.17 and 4.51 mm, i.e. 51.9% and 56% of the reported maxima, respectively (*Figure 8*). As a result, it is possible to conclude that ANN model M7-H1 is best suited for projecting accurate streamflows of Tessa watershed.

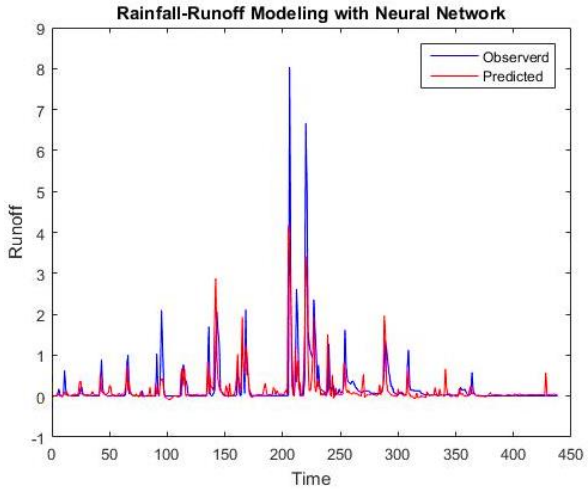


***Figure 6.*** *Comparison between simulated and observed flow using the M7 model*

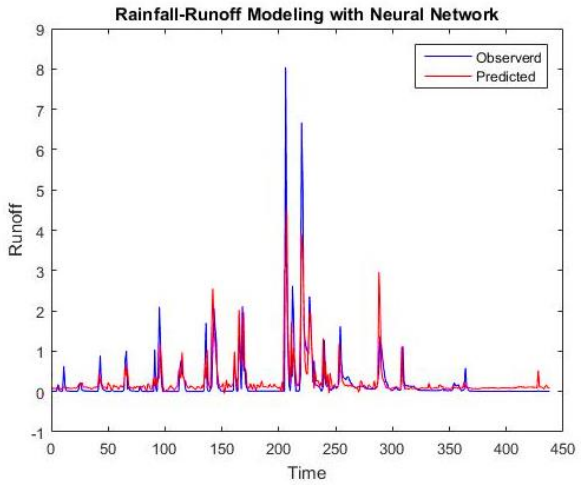


***Figure 7.*** *Comparison between simulated and observed flow using the M7-H1 model*

The effect of the database division function was also investigated; initially, this function was activated for a chronological block division, as mentioned at the outset of this study, and then we investigated the effect of allowing the program to divide the database randomly while maintaining the proportions of 70%, 15%, and 15% for training, validation, and testing, respectively. We used the initial M7 and M7-H1 models for this section of the investigation. *Table 7* compares the results of chronological and random block division to highlight the variety of metric parameters in this investigation. These


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

results showed an improvement in the accuracy of the two models employed, with the M7-H1-DR (M7-H1-Divide Randomly) model achieving a maximum NSE of 0.6349. However, it should be highlighted that the database's random division method does not reveal whether the 2003 floods were used as a training, validation, or test period.

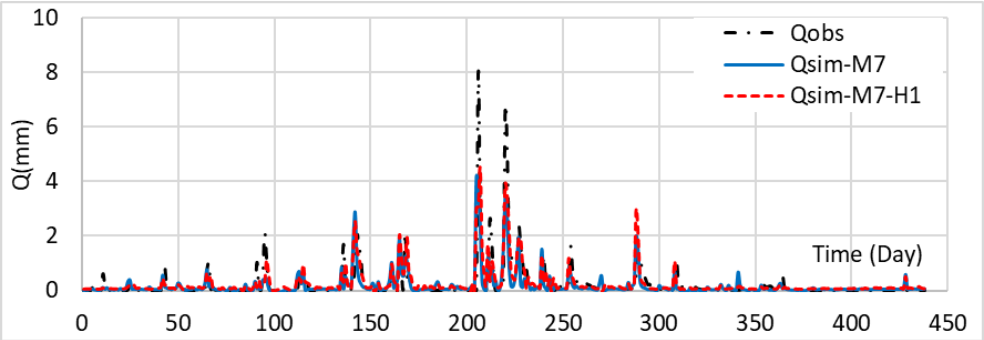


***Figure 8.*** *Comparison between observed and simulated flow using the M7-H1 model*

***Table 7.*** *Effect of the division function on the results*

| | **M7 (Divide by block)** | **M7-DR (Divide Randomly)** | **M-7-H1 (Divide by block)** | **M7-H1-DR (Divide randomly)** |
|---|---|---|---|---|
| NSE | 0.4786 | 0.6102 | 0.52 | 0.6349 |
| RMSE | 0.4865 | 0.6037 | 0.4667 | 0.5666 |
| MAE | 0.1903 | 0.1343 | 0.2065 | 0.175 |

## Discussion

In this research, the network exhibited a remarkably high level of accuracy during the training phase. The evaluation of the ANN model involved comparing simulated and observed hydrographs. Sensitivity analysis of the inputs revealed that various factors contribute to the generation of runoff, and rainfall values alone do not represent the sole variable influencing this process. Evapotranspiration is also an important input parameter. As shown in *Figure 3*, the input layer interacts with the external environment to collect data, which is subsequently passed on to the hidden levels for processing. After being processed in the hidden layers, this data is delivered to the outer layers, which expose results to the external world based on the information obtained from the hidden levels (Xua et al., 2008).

Choosing the right number of neurons in hidden layers and the size level of the hidden layers can be challenging due to overfitting and underfitting, which can negatively impact network efficiency and time complexity (Panchal et al., 2014).

Underfitting takes place when the number of hidden layers in a network is lacking to handle the complexity of the problem (Zhang et al., 2018). This is sometimes referred to as undertraining. This significantly reduces network efficiency. In such cases, the network's temporal complexity decreases, leading to inefficient results (Karsoliya et al., 2012).

Overfitting occurs when the number of hidden layers exceeds the complexity of the problem, resulting in a process of overtraining of the network, which has a negative impact on the network's time complexity. This usually happens when the network closely


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

aligns with the training data, causing a loss of its ability to generalize effectively over the test data (Asthana et al., 2017).

According to Awan (2018), great accuracy can only be attained if the network understands the entire problem with no overfitting or underfitting situations. Neural networks require a significant number of hidden layers and neurons for optimal results. *Figure 9* presents the results for the different structures of the M7 model, in terms of RMSE, and we note that the minimum value of the RMSE was obtained for architecture [4-3-1], i.e. for a number of neurons equal to three for the hidden layer. For [4-4-1], [4-5-1], [4-7-1], [4-8-1] and [4-9-1] ANN Architectures, RMSE values are less than 0.6, while ANN architectures [4-2-1], [4-10-1] and have an RMSE value close to 1.6 and 1 respectively, which means that for this structure, the model is underfitting. Extending the number of hidden neurones at the single hidden layer demonstrates that, as *Figure 9* illustrates, having more than three nodes for the single hidden layer guarantees stable performance.

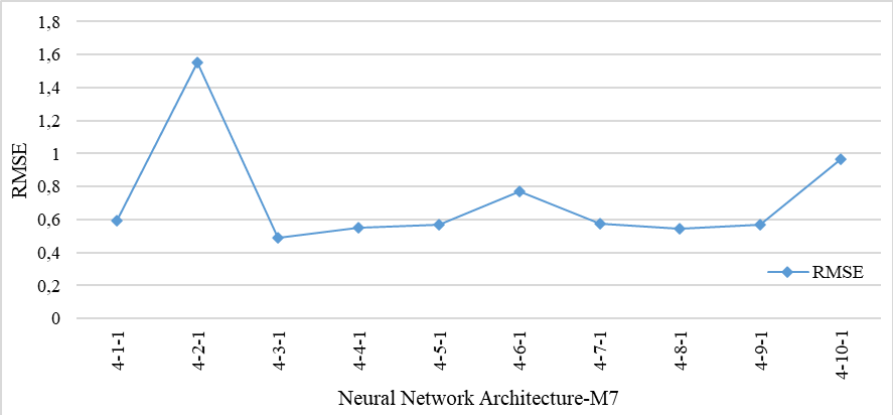


***Figure 9.*** *Relationship between RMSE and the number of hidden layers for the M7 model*

To determine the ideal number of hidden layers, we analyzed the patterns in RMSE while varying the number of hidden layers. The validation set, which was previously sampled independently from the learning data, is used to process the RMSE values. The results presented in *Table 8* and *Figure 10* indicate that the RMSE for 2 hidden layers ANN architecture, becomes steady after a particular point but for 3 hidden layers architecture the RMSE is more fluctuating.

***Table 8.*** *Number of the hidden layer effects on the results*

| Architecture | M7-H-2levels | | M7-H-3levels | |
|---|---|---|---|---|
| ID | Structure | RMSE | Structure | RMSE |
| 1 | 4-3-1-1 | 0,4667 | 4-3-1-1-1 | 0,5361 |
| 2 | 4-3-2-1 | 0,665 | 4-3-1-2-1 | 0,6394 |
| 3 | 4-3-3-1 | 1,4412 | 4-3-1-3-1 | 0,6801 |
| 4 | 4-3-4-1 | 0,5197 | 4-3-1-4-1 | 0,7641 |
| 5 | 4-3-5-1 | 0,5437 | 4-3-1-5-1 | 1,0239 |
| 6 | 4-3-6-1 | 0,5105 | 4-3-1-6-1 | 0,7173 |
| 7 | 4-3-7-1 | 0,5843 | 4-3-1-7-1 | 1,7667 |
| 8 | 4-3-8-1 | 0,5769 | 4-3-1-8-1 | 0,6371 |
| 9 | 4-3-9-1 | 0,5287 | 4-3-1-9-1 | 0,6794 |
| 10 | 4-3-10-1 | 0,5747 | 4-3-1-10-1 | 1,3684 |


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

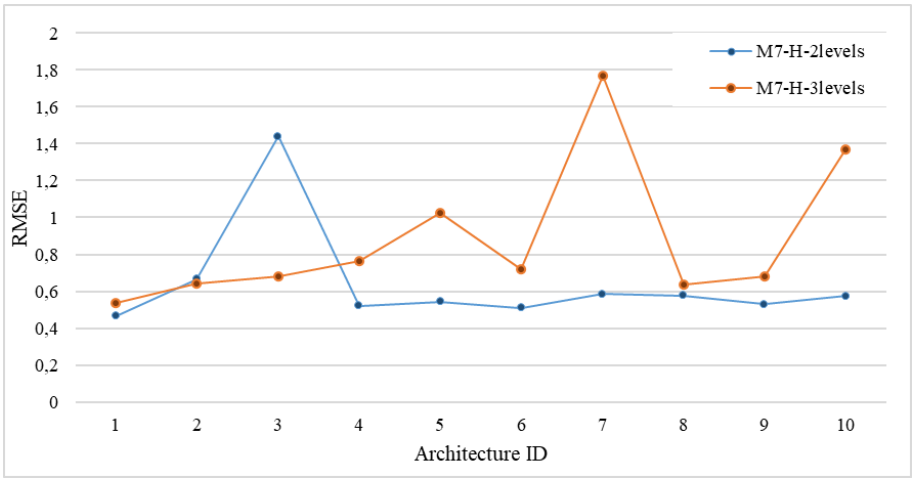


***Figure 10.*** *Relationship between RMSE and the number of hidden layers for the M7 model*

It indicates that when a sufficient number of hidden layers are applied, we reach improved results with a much lower time complexity. On the contrary, increasing the number of hidden layers can lead to significantly improved accuracy, but it also results in a more complex neural network compared to the previous one. This observation was also reported by Asthana et al. (2017).

## Conclusion

In this study, nine ANN models with various combinations of input variables were constructed and implemented to improve the predicting accuracy of daily streamflows in the Tessa watershed in Tunisia's semiarid region. Daily rainfall recorded at 12 rainfall stations from 01/09/1995 to 31/08/2003, as well as evaporation and daily flow recorded at the Sidi Meddien hydrometric station, were used as input variables. In this study, the effect of the length of the database allocated to training the model was studied first, and the best results were found with 70% of the dataset, and the remaining data was used for validation and testing. The effect of the number of neurons and the level of the hidden layer have also been investigated. The results reveal that ANN models M7 and M7-H1 exhibit significant fluctuation in terms of NSE values (0.4786 and 0.52) over both the calibration process and validation periods. Moreover, the input vectors P(t), EVT(t), EVT(t-1), and Q(t-1) outperformed the other input vectors. When compared to observed streamflows, the results demonstrate that the M7 and M7-H1 models replicate the peak flows on the same date, but with significantly lower values 51.9% and 56% of the recorded maxima, respectively. As a result, although underestimating the peak flow, the ANN model effectively successfully captures the intricate and non-linear structure of hydrologic time series, providing accurate and reliable streamflow forecasts in a semi-arid region.

## REFERENCES

[1] Adeyemo, J., Oyebode, O., Stretch, D. (2018): River flow forecasting using an improved artificial neural network. – In: Tantar, A. A., Tantar, E., Emmerich, M., Legrand, P., Alboaie, L., Luchian, H. (eds.) EVOLVE - A Bridge between Probability, Set Oriented


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

Numerics, and Evolutionary Computation VI. Advances in Intelligent Systems and Computing, Springer, Cham. vol 674.
[2] Alp, M., Cigizoglu, H. K. (2007): Suspended sediment estimation by feed forward back propagation method using hydrometeorological data. – Environ. Model. Software 22(1): 2-13.
[3] Asrhana, A., Pandit, A., Bahrdwaj, A. (2017): Review on Methods of selecting Number of Hidden Nodes in Artificial Neural Network. – IJCSMC 3(11): 455-464.
[4] Awan, A., Riaz, M., Khan, A. (2018): Prediction of Heart Disease Using Artificial Neural network. – VFAST Transaction on Software Engineering 13.
[5] Chen, S. M., Wang, Y. M., Tsou, I. (2013): Using artificial neural network approach for modelling rainfall runoff due to typhoon. – J Earth Syst Sci 122(2): 399-405.
[6] Cigizoglu, H. K. (2003a): Incorporation of ARMA models into flow forecasting by artificial neural networks. – Environmetrics 14(4): 417-427.
[7] Cigizoglu, H. K. (2003b): Estimation, forecasting and extrapolation of flow data by artificial neural networks. – Hydrol. Sci. J. 48(3): 349-361.
[8] Dakhlaoui, H., Bargaoui, Z., Bárdossy, A. (2012): Toward a more efficient calibration schema for HBV rainfall–runoff model. – J Hydrol 444: 161-179.
[9] de O. Galvão, C. (1999): Sistemas Inteligentes: aplicações a recursos hídricos e ciências ambientais. – Porto Alegre, Brazil: Universidade/ UFRGS/ABRH.
[10] Ebtehaj, I., Bonakdari, H., Zaji, A. H., Gharabaghi, B. (2020): Evolutionary optimization of neural network to predict sediment transport without sedimentation. – Complex Intell Syst 7(1): 401-416.
[11] Fernandes, L. G. L., Navaux, P. O. A., Portugual, M. S. (1996): Previsão de séries de tempo: redes neurais artificiais e modelos estruturais. – Pesquisa e Planejamento Econômico PPE 26(2): 253-276.
[12] Ghazi, B., Jeihouni, E., Kalantri, Z. (2021): Predicting groundwater level fluctuations under climate change scenarios for Tasuj plain. – Iran Arab J Geosci 14: 2.
[13] Jain, S. K., Das, A., Srivastava, D. K. (1999): Application of ANN for reservoir inflow prediction and operation. – J. Water Resour. Plan. Manage. ASCE 125(5): 263-271.
[14] Jain, A., Prasad Indurthy, S. K. V. (2003): Comparative analysis of event-based rainfall-runoff modeling techniques-deterministic, statistical, and artificial neural networks. – J Hydrol Eng 8(2): 93-98.
[15] Karasoliya, S. (2012): Approximating Number of Hidden layer Neurons in Multiple hidden Layer BPNN Architecture. – International Journal of Engineering Trends and Technology 3(6).
[16] Khan, M. Y. A., Hasan, F., Panwar, S., Chakrapani, G. J. (2016): Neural network model for discharge and water-level prediction for Ramganga River catchment of Ganga Basin, India. – Hydrol Sci J 61(11): 2084-2095.
[17] Lu, P., Chen, S., Zheng, Y. (2012): Artificial intelligence in civil engineering. – Mathematical Problems in Engineering 2012: 145974.
[18] Machado, F., Mine, M., Kaviski, E., Fill, H. (2011): Monthly rainfall–runoff modelling using artificial neural networks. – Hydrological Sciences Journal 56(3): 349-361.
[19] May, R. J., Dandy, G. C., Maier, H. R., Nixon, J. B. (2008): Application of partial mutual information variable selection to ANN forecasting of water quality in water distribution systems. – Environ Model Softw 23(10-11): 1289-1299.
[20] Mins, A. W., Hall, M. J. (1996): Artificial neural network as a rainfall runoff models. – Hydrol Sci J 41(3): 399-417.
[21] Nagy, H. M., Watanabe, K., Hirano, M. (2002): Prediction of sediment load concentration in rivers using artificial neural network model. – J Hydr Eng 128: 588-595.
[22] Neelakantan, T. R., Pundarikanthan, N. V. (2000): Neural network-based simulation-optimization model for reservoir operation. – J. Water Resour. Plan. Manage. ASCE 126(2): 57-64.


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998

[23] Panchal, F., Panchal, M. (2014): Review on methods of selecting number of hidden nodes in artificial neural network. – IJCSMC 3(11): 455-464.
[24] Poonia, V., Tiwari, H. L. (2020): Rainfall-runoff modeling for the Hoshangabad Basin of Narmada River using artificial neural network. – Arab J Geosci 13(18): 1-10.
[25] Rezaeianzadeh, M., Stein, A., Tabari, H., Abghari, H., Jalalkamali, N., Hosseinipour, E. Z., Singh, V. P. (2013): Assessment of a conceptual hydrological model and artificial neural networks for daily outflows forecasting. – Int J Environ Sci Technol 10(6): 1181-1192.
[26] Sharma, P., Madane, D., Bhakar, S. R., Sharma, S. D. (2021): Monthly streamflow forecasting using artificial intelligence approach: a case study in a semi-arid region of India. – Arab J Geosci 14: 2440.
[27] Wagena, M. B., Goering, D., Collick, A. S., Bock, E., Fuka, D. R., Buda, A., Easton, Z. M. (2020): Comparison of short-term streamflow forecasting using stochastic time series, neural networks, process-based, and Bayesian models. – Environ Model Softw 126: 104669.
[28] Warwade, P., Tiwari, S., Ranjan, S., Chandniha, S. K., Adamowski, J. (2018): Spatio-temporal variation of rainfall over Bihar State, India. – J Water Land Dev 36(1): 183-197.
[29] Wu, W., Dandy, G. C., Maier, H. R. (2014): Protocol for developing ANN models and its application to the assessment of the quality of the ANN model development process in drinking water quality modeling. – Environ Model Softw 54(2014): 108-127.
[30] Xua, S., Chen, L. (2008): A novel approach for determining the optimal number of hidden layer neurons for FNN's and its application in data mining. – 5th International Conference on Information Technology and Applications (ICITA 2008).
[31] Zhang, Q., Zhu, S-C. (2018): Visual Interpretability for Deep learning. – arXiv: 1802.00614 V2.
[32] Zhang, B., Govindaraju, R. S. (2000): Prediction of watershed runoff using Bayesian concepts and modular neural networks. – Water Resour Res 36(3): 753-762.


http://www.aloki.hu ● ISSN 1589 1623 (Print) ● ISSN 1785 0037 (Online)
DOI: http://dx.doi.org/10.15666/aeer/2206_49834998