environmental sciences proceedings

MDPI

Proceeding Paper

Rainfall-Runoff Modeling Using Artificial Neural Network—A Case Study of Purna Sub-Catchment of Upper Tapi Basin, India †

Usman Mohseni * and Sai Bargav Muskula

Department of Civil Engineering, IIT Roorkee, Roorkee 247667, India; muskula_sbr@ce.iitr.ac.in
* Correspondence: mohseni_ua@ce.iitr.ac.in

† Presented at the 7th International Electronic Conference on Water Sciences, 15–30 March 2023; Available online: https://ecws-7.sciforum.net/.

Abstract: The present study examines the rainfall-runoff-based model development by using artificial neural networks (ANNs) models in the Yerli sub-catchment of the upper Tapi basin for a period of 36 years, i.e., from 1981 to 2016. The created ANN models were capable of establishing the correlation between input and output data sets. The rainfall and runoff models that were built have been calibrated and validated. For predicting runoff, Feed-Forward Back Propagation Neural Network (FFBPNN) and Cascade Forward Back Propagation Neural Network (CFBPNN) models are used. To evaluate the efficacy of the model, various measures such as mean square error (MSE), root mean square error (RMSE), and coefficient of correlation (R) are employed. With MSE, RMSE, and R values of 0.4982, 0.7056, and 0.96213, respectively, FFBPNN outperforms two networks with model architectures of 6-4-1 and Transig transfer function. Additionally, in this study, the Levenberg–Marquardt (LM), Bayesian Regularization (BR), and Conjugate Gradient Scaled (CGS) algorithms are used to train the ANN rainfall-runoff models. The results show that LM creates the most accurate model. It performs better than BR and CGS. The best model is the LM-trained method with 30 neurons, which has MSE values of 0.7279, RMSE values of 0.8531, and R values of 0.95057. It is concluded that the constructed neural network model was capable of quite accurately predicting runoff for the Yerli sub-catchment.

Keywords: artificial neural network; rainfall-runoff modeling; feed-forward back propagation; cascade forward back propagation

Citation: Mohseni, U.; Muskula, S.B. Rainfall-Runoff Modeling Using Artificial Neural Network—A Case Study of Purna Sub-Catchment of Upper Tapi Basin, India. *Environ. Sci. Proc.* **2023**, *25*, 1. https://doi.org/10.3390/ECWS-7-14232

Academic Editor: Athanasios Loukas

Published: 16 March 2023

1. Introduction

Hydrologists have been attempting to understand the translation of rainfall to runoff for many years to estimate streamflow for objectives including water supply, flood control, irrigation, drainage, water quality, power production, recreation, and fish and wildlife propagation [1]. Rainfall-runoff modeling is one of the most prominent hydrological models used to examine the relationship between rainfall and runoff generated by various watershed physical factors [2]. In the real world, all physical catchment features influence rainfall-runoff; hence, generalizing all physical catchment characteristics is a difficult process. It is difficult to depict such a large range of values in a lumped hydrological model since the parameter values must be averaged for each watershed [3].

In the past, academics and hydrologists have presented different ways for effectively forecasting runoff by building several models of rainfall-runoff (RR) [4]. The process of rainfall-runoff is highly nonlinear and incredibly complex and is still poorly understood [5]. Furthermore, several rainfall-runoff models require a substantial amount of data, which are used for calibration and validation time scale, making them computationally intensive and, thus, unpopular [6]. Machine learning techniques are becoming more prevalent due to their ease of use, simplicity, and efficiency [7]. Machine learning techniques are a good option when there are minimal data and the process is complex [8]. In the context of estimating issues, artificial neural networks (ANNs) are a subclass of machine learning that have received significant attention [9]. ANNs are data-processing systems that mimic the capabilities of the human brain [10]. ANNs were first developed in the 1940s and come in a wide variety [11]. ANN models are also known as black-box models [12]. The application of ANNs in the creation of models results in trustworthy and versatile learning ability, which makes ANNs promising for forecasting [13]. ANN models have been extremely prevalent in the domains of hydrology, water resources, and watershed management in recent decades [14]. The ANN contains three layers, an input layer, a hidden layer, and an output layer [15]. The weight of communication is the relationship between the neurons in the consecutive layers [16]. In the given study, the input layer consists of six types of data, namely (rainfall, minimum temperature, maximum temperature, surface pressure, specific humidity, and wind speed). The hidden layer consists of layers with two different sets of number of neurons 10 and 20, respectively. The output layer comprises predicted runoff.

The objectives of the present study are as follows: (i) to develop a rainfall-runoff model for Upper Tapi using an Artificial Neural Networks Technique, (ii) compare ANN rainfall-runoff models developed using NNTOOL with different neural network types, i.e., FFBPNN and CFBPNN, and (iii) to compare ANN rainfall-runoff models trained using LM, BR, and SCG algorithms.

2. Materials and Methods

2.1. Study Area and Data Collection

The current study area comprises a portion of the Upper Tapi Basin known as the Purna sub-catchment (Figure 1). The area lies between Maharashtra and Madhya Pradesh, between latitudes of 20°09′ N to 22°03′ N and longitudes of 75°56′ E to 78°17′ E. It has subtropical to temperate climatic conditions. The mean annual precipitation in the chosen area varies from 833 to 990 mm. Table 1 reveals the sources of data for this study.

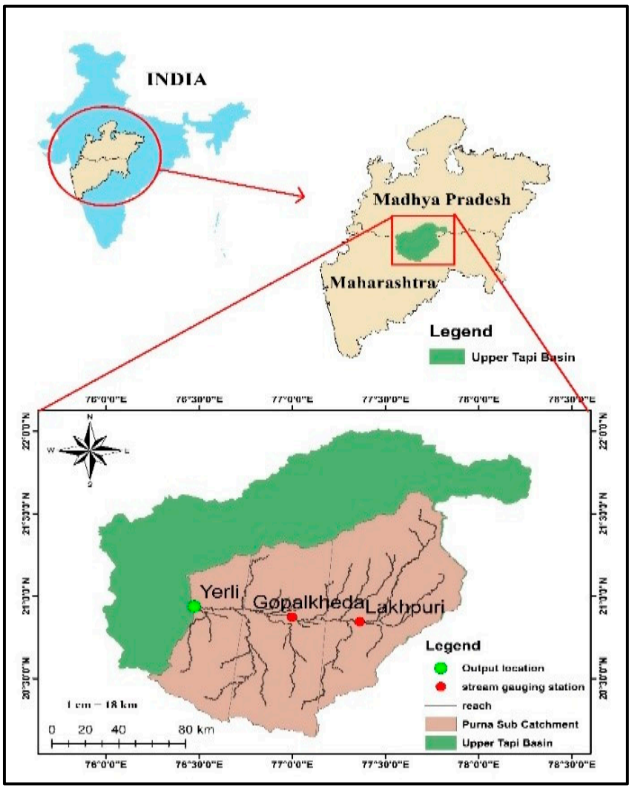

Figure 1. Index Map of the study area.

Table 1. Source of data.

Data Type	Data Source
Digital Elevation Model	USGS Earth Explorer
Rainfall	Central Water Commission
Meteorological data	India Meteorological Department
Discharge	Central Water Commission

2.2. Methodology

The theoretical aspects and research methodology used in the current study to identify the best neural network model to perform the rainfall and runoff modeling for the Yerli sub-catchment have been discussed in this section. Figures 2 and 3 depict the methodological flowchart of NNTOOL and NNSTART respectively.

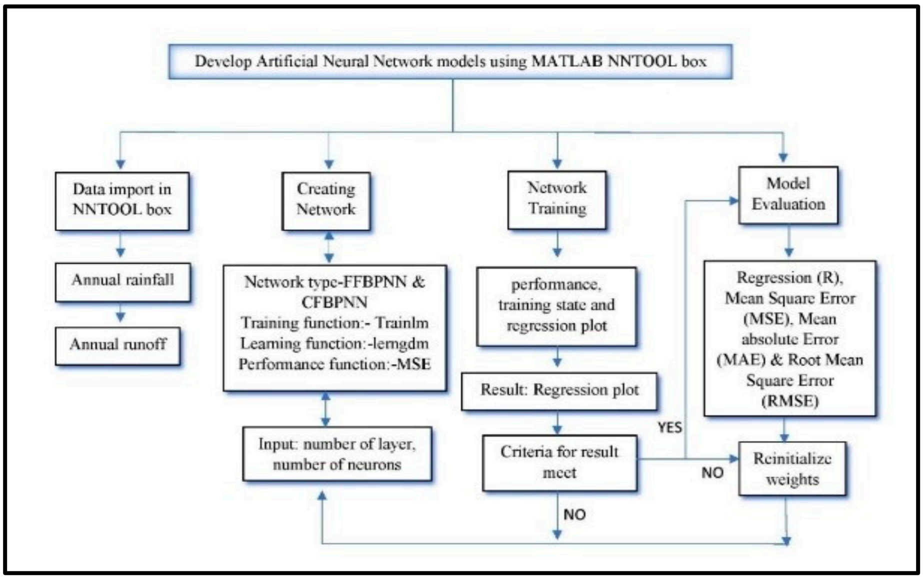

Figure 2. Flow chart of NNTOOL.

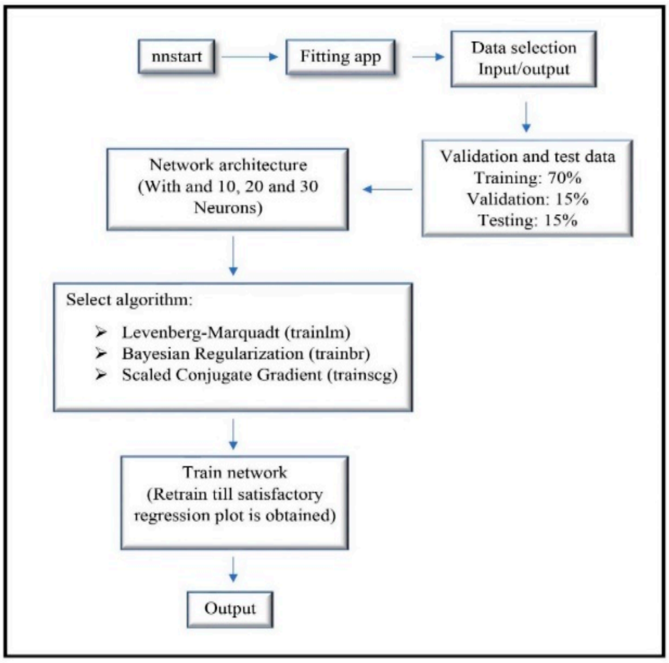

Figure 3. Flow chart of NNSTART.

2.2.1. Following Steps Should Be Performed for Developing an ANN Model Using NNTOOL

- Data Collection: The required observed data (rainfall, runoff, temperature, specific humidity, surface pressure, wind speed) at the prerequisite station are to be collected.
- Import Data: The collected data are imported into the NNTOOL box as input and target data.
- Creating Network: The network is created by selecting a suitable network type, i.e., FFBPNN or CFBPNN. The network architecture is formed (6-2-1, 6-3-1, 6-4-1).
- Number of Neurons: For the given network, the number of neurons is taken as 10 or 20.
- Network Training: The developed network is trained based on performance function.
- Result: Once the network is trained, the result is checked by plotting the regression plot, and the predicted output is obtained.
- Retraining: If the obtained regression plot is not satisfactory, then reinitialization of weights has to be conducted by changing the number of neurons.
- Model Evaluation: Based on statistical parameters such as (MSE), (RMSE), (R^2), and (R).

2.2.2. Following Steps Should Be Performed for Developing an ANN Model Using NNSTART

- Neural Fitting App: The Neural Fitting app will help to select data, create and train a network, and evaluate its performance using mean square error and regression analysis.
- Data Selection: The collected data will be used as both the input and output data. The input data are in a 6×36 matrix. On the other hand, the target data are in a 1×36 matrix.
- Validation and Test: The data are split as follows, 70% (training), 15% (validation), and 15% (testing).
- Network Architecture: For the given network, the number of neurons is taken as 10, 20, and 30.
- Select Algorithm: For training, the algorithms, namely Levenberg–Marquardt (trainlm), Bayesian Regularization (trainbr), and Scaled Conjugate Gradient (trainscg), were used.
- Train Network: To fit the input and goal data, train the network.
- Retrain: The network is retrained if a satisfactory regression plot is not obtained.
- Output: Desired predicted output is obtained after fixing the regression plot.

2.3. Model Evaluation Criteria

The findings of the ANN model applied in this study were evaluated by means of:

- Mean Square Error (MSE):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(Q_p - Q_o\right)^2 \quad (1)$$

- Root Mean Square Error (RMSE):

$$\text{RMSE} = \left[\frac{\sum_{i=1}^{n}\left(Q(i) - \hat{Q}(i)\right)^2}{n}\right]^{0.5} \quad (2)$$

- Regression Coefficient (R): Using Regression Plot between predicted and observed runoff.

where Q_p is the value of predicted runoff; Q_o is the value of observed runoff; $\hat{Q}(i)$ is the n estimated runoff value; and $Q(i)$ is the n observed runoff value.

3. Results and Discussion

3.1. NNTOOL

The multilayer FFBPNN and CFBPNN algorithms with Levenberg–Marquardt (LM) are utilized to optimize the learning approach in this study. Two different models were developed, i.e., (FFBPNN) and (CFBPNN) with three different architectures (6-2-1, 6-3-1 and 6-4-1) using several combinations of transfer functions, i.e., (transig, logsig, and purelin) along with two sets of neurons, 10 and 20, and then compared for their capability to estimate the flow for the period 1981–2016.

3.1.1. Feed Forward Back Propagation Neural Network (FFBPNN)

FFBPN, while considering 6-2-1, 6-3-1, and 6-4-1 architectures, the transig function provides the best value for performance. The most effective model architecture for the Transig function is 6-4-1, which has a value of MSE 0.4982, the value of RMSE 0.7056, and the value of R 0.96213. Table S1 contains the inclusive outcomes. However, in comparison to other transfer functions, the transig transfer function with architecture 6-4-1 yielded better results in the current study. Figure 4 depicts the best regression plot.

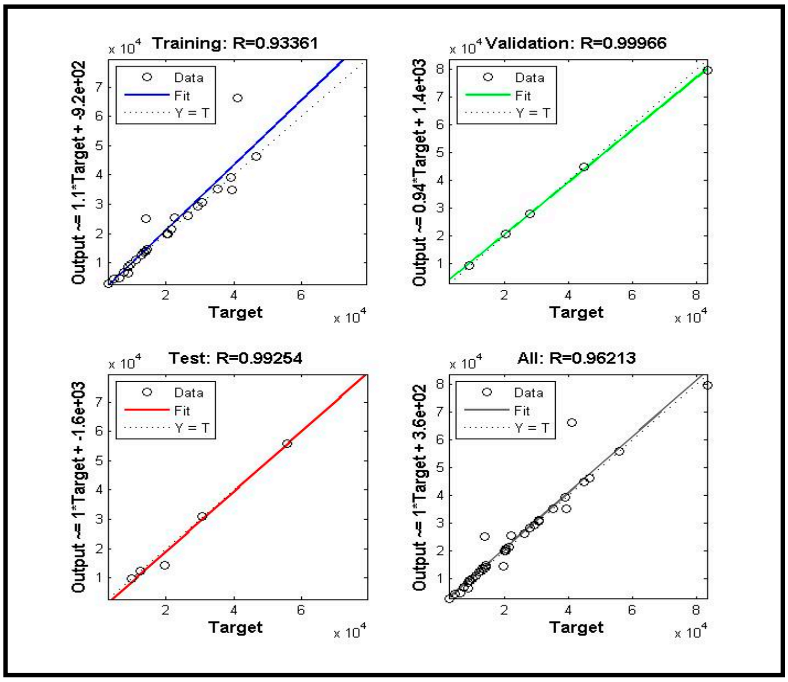

Figure 4. Regression plot for FFBPNN 6-4-1 model.

3.1.2. Cascade Forward Back Propagation Neural Network (CFBPNN)

Similarly, for CFBPNN, while considering 6-2-1, 6-3-1, and 6-4-1 architectures, the transig function provides the best value for performance. The most effective model architecture for the transig function is 6-4-1, which has MSE values of 0.8813, the value of RMSE 0.9387, and the value of R 0.96096. Table S2 contains the inclusive outcomes [17]. However, in comparison to other transfer functions comparison to other, the transig transfer function with architecture 6-4-1 yields better results. Figure 5 depicts the best regression plot.

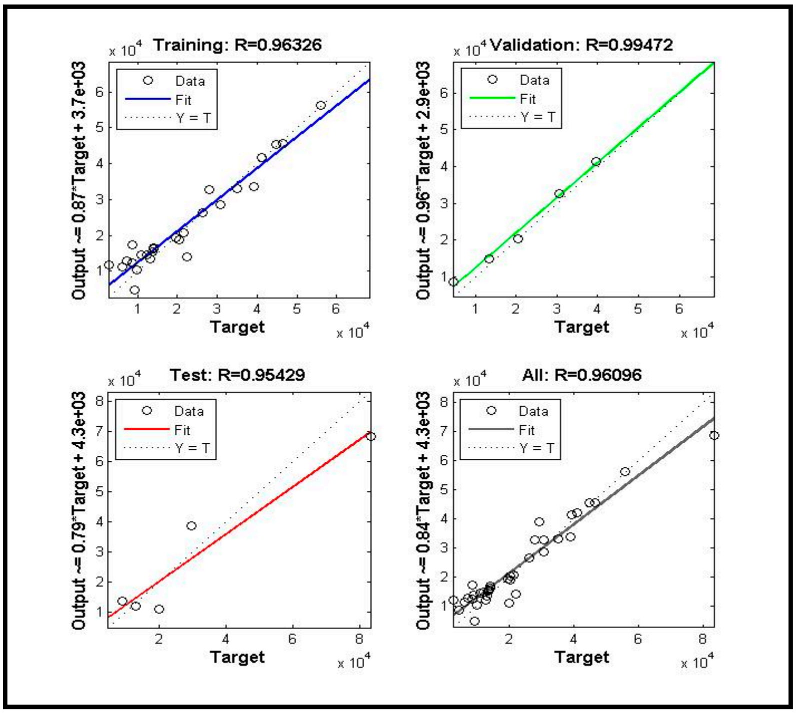

Figure 5. Regression plot for CFBPNN 6-4-1 model.

3.2. NNSTART

In this study, three different algorithms, namely Levenberg–Marquardt (trainlm), Bayesian Regularization (trainbr), and Scaled Conjugate Gradient (trainscg), were used for model development. Table S3 shows the Yerli station results for the ANN trained by LM, BR, and CGS. The study compares ANN models that were trained with LM, BR, and CGS. LM-trained algorithm with 30 neurons is the best model with an MSE value of 0.7279, an RMSE value of 0.8531, and an R value of 0.95057. Figure 6 shows the best regression plot for the LM algorithm with 30 neurons.

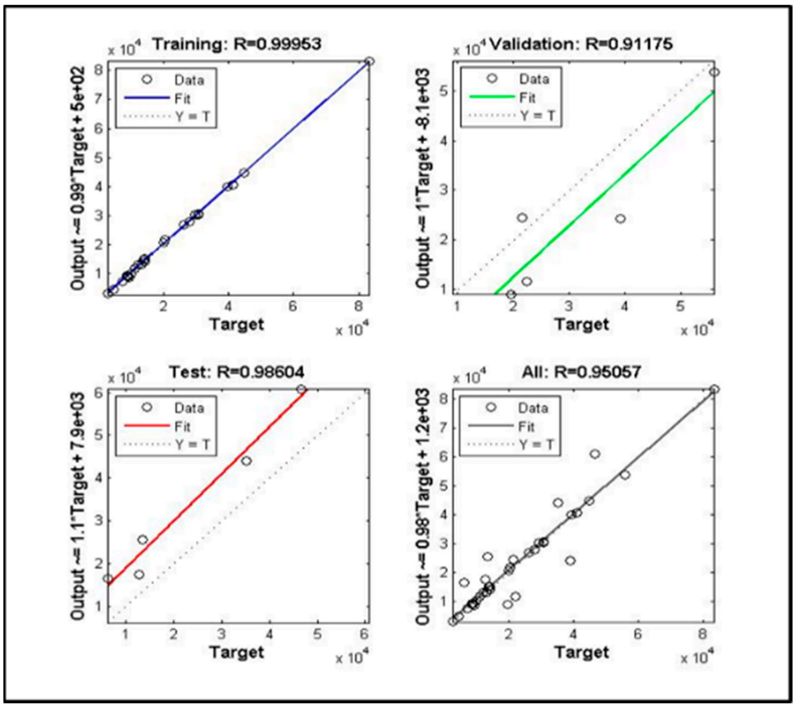

Figure 6. Regression plot for LM algorithm with 30 neurons.

4. Conclusions

This study described how ANN models are used to estimate yearly runoff for the Yerli sub-catchment of the upper Tapi basin. Runoff estimation was undertaken using NNTOOL and NNSTART. Adopting NNTOOL, two different models were developed, i.e., FFBPNN and CFBPNN networks, using several combinations of input data and then comparing their capability of flow estimation for the period 1981–2016. For estimating runoff using NNTOOL, two NNs are used, with the values of MSE, RMSE, and R calculated. For the transig function in FFBPNN, the most prominent model architecture is 6-4-1, which has an MSE value of 0.4982, an RMSE value of 0.7056, and a value of R of 0.96108. The 6-4-1 model architecture for the transig function is the most effective for CFBPNN, with MSE values of 0.8813, RMSE values of 0.9387, and R values of 0.96096. Using three different algorithms, LM, BR, and CGS, were used to predict runoff. Among the three, the LM-trained algorithm with 30 neurons is the best model, with MSE values of 0.7279, RMSE values of 0.8531, and R values of 0.95057. According to the findings, FFBPNN predicts better results than CFBPNN, and the LM algorithm stands out among the other algorithms.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/ECWS-7-14232/s1, Table S1: Results of FFBPNN for Yerli station, Table S2: Results of CFBPNN for Yerli station, Table S3: Results of NNSTART for Yerli station.

Author Contributions: Conceptualization, U.M.; methodology, U.M.; software, U.M.; validation, U.M.; formal analysis, U.M. and S.B.M.; data curation, U.M.; writing—original draft preparation, U.M.; writing—review and editing, U.M. and S.B.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Unavailability of data due to ethical restrictions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tokar, A.S.; Johnson, P.A. Rainfall-runoff modeling using artificial neural networks. *J. Hydrol. Eng.* **1999**, *4*, 232–239. [CrossRef]
2. Namara, W.G.; Damise, T.A.; Tufa, F.G. Rainfall Runoff Modeling Using HEC-HMS: The Case of Awash Bello Sub-Catchment, Upper Awash Basin, Ethiopia. *Int. J. Environ.* **2020**, *9*, 68–86. [CrossRef]
3. Rathod, P.; Borse, K.; Manekar, V.L. Simulation of rainfall-runoff process using HEC-HMS (case study: Tapi river, India). In Proceedings of the 20th International Conference on Hydraulics, Water Resources and River Engineering, Roorkee, India, 17–19 December 2015.
4. Chen, J.; Adams, B.J. Integration of artificial neural networks with conceptual models in rainfall-runoff modeling. *J. Hydrol.* **2006**, *318*, 232–249. [CrossRef]
5. Zhang, B.; Govindaraju, R.S. Prediction of watershed runoff using Bayesian concepts and modular neural networks. *Water Resour. Res.* **2000**, *36*, 753–762. [CrossRef]
6. Lu, P.; Chen, S.; Zheng, Y. Artificial intelligence in civil engineering. *Math. Probl. Eng.* **2012**, *2012*, 145974. [CrossRef]
7. Tayyab, M.; Zhou, J.; Adnan, R.; Zeng, X. Application of Artificial Intelligence Method Coupled with Discrete Wavelet Transform Method. *Procedia Comput. Sci.* **2017**, *107*, 212–217. [CrossRef]
8. Chae, Y.T.; Horesh, R.; Hwang, Y.; Lee, Y.M. Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings. *Energy Build.* **2016**, *111*, 184–194. [CrossRef]
9. Khayatian, F.; Sarto, L.; Dall'O', G. Application of neural networks for evaluating energy performance certificates of residential buildings. *Energy Build.* **2016**, *125*, 45–54. [CrossRef]
10. Kisi, O.; Shiri, J.; Tombul, M. Modeling rainfall-runoff process using soft computing techniques. *Comput. Geosci.* **2013**, *51*, 108–117. [CrossRef]
11. Rogers, L.L.; Dowla, F.U. Optimization of groundwater remediation using artificial neural networks with parallel solute transport modeling has been successfully applied to a variety of optimization. *Water Resour. Res.* **1994**, *30*, 457–481. [CrossRef]
12. Vandana, M.; John, S.E.; Maya, K.; Sunny, S.; Padmalal, D. Environmental impact assessment (EIA) of hard rock quarrying in a tropical river basin—Study from the SW India. *Environ. Monit. Assess.* **2020**, *192*, 580. [CrossRef] [PubMed]

13. Sahour, H.; Gholami, V.; Vazifedan, M. A comparative analysis of statistical and machine learning techniques for mapping the spatial distribution of groundwater salinity in a coastal aquifer. *J. Hydrol.* **2020**, *591*, 125321. [CrossRef]
14. Orimi, M.G.; Farid, A.; Amiri, R.; Imani, K. Cprecip parameter for checking snow entry for forecasting weekly discharge of the Haraz River flow by artificial neural network. *Water Resour.* **2015**, *42*, 607–615. [CrossRef]
15. Chandwani, V.; Vyas, S.K.; Agrawal, V.; Sharma, G. Soft Computing Approach for Rainfall-runoff Modelling: A Review. *Aquat. Procedia* **2015**, *4*, 1054–1061. [CrossRef]
16. Chang, T.K.; Talei, A.; Quek, C.; Pauwels, V.R.N. Rainfall-runoff modelling using a self-reliant fuzzy inference network with flexible structure. *J. Hydrol.* **2018**, *564*, 1179–1193. [CrossRef]
17. Samantaray, S.; Sahoo, A. Prediction of runoff using BPNN, FFBPNN, CFBPNN algorithm in arid watershed: A case study. *Int. J. Knowl.-Based Intell. Eng. Syst.* **2020**, *24*, 243–251. [CrossRef]