Extreme Learning Machine with Evolutionary Parameter Tuning Applied to Forecast the Daily Natural Flow at Cahora Bassa Dam, Mozambique

Alfeu D. Martinho[1,2], Celso B. M. Ribeiro[3], Yulia Gorodetskaya[2], Tales L. Fonseca[2], and Leonardo Goliatt[2(✉)]

[1] Púnguè-Tete University, Chimoio, Mozambique
[2] Computational Modeling Program, Federal University of Juiz de Fora, Juiz de Fora, Brazil
goliatt@gmail.com
[3] Civil Engineering Program, Federal University of Juiz de Fora, Juiz de Fora, Brazil

Abstract. This paper proposes a hybrid approach combining an Extreme Learning Machine and a Genetic Algorithm to predict the short-term streamflow at the Cahora Bassa dam, the largest hydroelectric power plant in southern Africa. To predict the streamflows seven days ahead, the model uses as input the past river flows, information from humidity, rainfall, and evaporation measures from the lake upstream of the dam. The choice of the Extreme Learning Machine's internal parameters, crucial for excellent model performance, is performed by a Genetic Algorithm. A set of five metrics was used to assess the performance of the hybrid approach. The computational experiments show the proposed approach outperforms other machine learning methods such as ElasticNet linear model, Support Vector Machines, and Gradient Boosting. However, the ELM prediction model overestimates higher flows. The approach arises as a practical tool to predict the streams which have the potential to help the dam operations balancing the needs of energy production and the safety of the population living downstream of the dam.

Keywords: Extreme learning machines · Genetic algorithms · Hydrology

1 Introduction

Located in the province of Tete, the Cahora Bassa dam is the fourth-largest dam in Africa, on the terminal section of the so-called Zambezi medium Mozambique. The reservoir is the 12th largest in the world and also the fifth in Africa, with a maximum capacity of 6300 m^3 of water (after Aswan, Volta, and Kariba), has a maximum length of 250 km in length and 38 km of spacing between banks, occupying about 2700 km^2 and having an average depth of 26 m. It is currently the

B. Filipič et al. (Eds.): BIOMA 2020, LNCS 12438, pp. 255–267, 2020.
https://doi.org/10.1007/978-3-030-63710-1_20

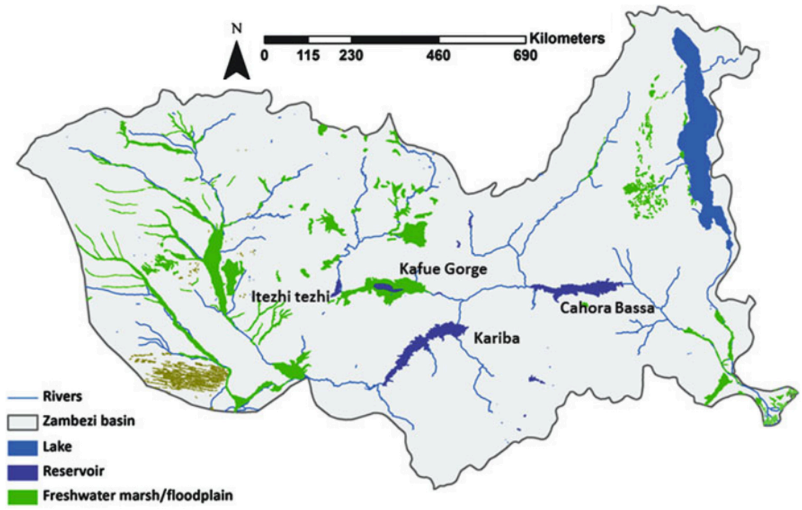

Fig. 1. Zambezi River basin, showing the location of the Cahora Bassa dam. Adapted from [16].

largest electricity producer in Mozambique, with over 2000 MW, which supplies Mozambique, South Africa, Zimbabwe, and Malawi.

In addition to energy production, it contributes to the development of the economy of the Zambezi River delta, downstream of it, by enabling activities such as agriculture, pastoralism, fishing, construction of access roads and, in reducing the risk of the occurrence of natural disasters, such as drought, floods. However, due to the maximization of energy production, by releasing stored water during the season drought, while preparing for low flows on dry season. As a result, the Zambezi's regulated flow has been drying out the wetlands, previously fed by the Zambezi floodwaters, ceasing to be multiple secondary channels and branches that regularly changed, becoming a single main channel. The water released by the dam erodes the banks and deepens the riverbed. The now dry floodplains have severe consequences for biodiversity. Floods have become unpredictable, making communities across the Zambezi much more vulnerable to their negative impacts. This paper presents a tool to help understand the dynamics of the Zambezi River in Cahora-Bassa, by forecasting the flow in the affluence to the dam. The Cahora Bassa lake receives contributions from the Luangwa River Sub-basin, Kafue River Sub-basin, Hunyani River Sub-basin, and direct tributaries to the reservoir and by the effluence generated by Kariba, the upstream dam, depending on its storage status. This forecast appears to be extremely important because it can help make decisions about dam operations, contributing to the construction of a policy of sustainable exploitation of the hydrographic basin, that is, the adoption of the ecological flow. Figure 1 shows the Zambezi River basin and the location of the Cahora Bassa dam.

Time series prediction models have been widely applied for the characterization of hydrological variables. Several methods developed in the literature are used to this task, ranging from hydrological and meteorological modeling, statistical approaches, and computational intelligence and machine learning models. As reported in the recent literature [1,15,20] the machine learning models deserve a particular highlight in their modeling and forecasting abilities, often obtaining better results.

A comparison of machine learning techniques for monthly river flow forecasting is reported in [13] and [21]. By comparing eight machine learning models for time series prediction, Ahmed et al. [2] have concluded that multilayer perceptron neural networks and Gaussian processes produced the most accurate estimations. Sun et al. [19] reported that Gaussian Process outperformed ARIMA-based methods in more than 400 river basins. Other computational intelligence approaches have also have shown to be accurate tools for water level and discharge forecasting with uncertainty, such as Fuzzy neural networks [3].

Despite the impressive results reported in the literature, machine learning approaches' proper performance depends on adjusting internal parameters [14], and their choice directly affects the performance of the models. For example, neural networks need the number of layers, the number of neurons, and the learning rate to be set. Gaussian processes need the choice of the kernel function and the associated parameters. This task can exhibit high complexity [6], and smart search techniques are an alternative to find the best possible set. To tackle these drawbacks, an alternative is to apply an optimization algorithm. This approach can be called hybrid. Hybrid strategies combine the capabilities of different methods to produce accurate predictions, covering physical modeling, concentrated and distributed conceptual models, methodologies combining both machine learning and stochastic models, and techniques of artificial intelligence and data mining.

This study aims to assess the abilities of the Extreme Learning Machine (ELM) model as a practical technique for predicting the natural flow of the Zambezi River to the Cahora-Bassa dam. The results show that ELM outperforms other machine learning methods producing accurate estimations. However, the technique should be applied with care when estimating extreme flow values. The ELM can be used to estimate natural flows helping to develop an alert system for the dam's operations. This paper is organized follows. Section 2 describes the study area and the historical data, the streamflow estimation model, and the proposed hybrid approach. The computational experiments and discussion are presented in Sect. 3. Finally, Sect. 4 draws the conclusion.

2 Materials and Methods

2.1 Study Area and Data

The data consist of a historical daily series of 5844 observations covering 15 years, referring to 2003 and 2018. The variables under analysis are: natural flow affluent to the Cahora Bassa dam (Q), rainfall (R), evaporation (E) and relative

humidity (H). The database provided by the Water Resources and Environment Management Department of the Cahora Bassa Hydroelectric, the managing company of the Cahora Bassa dam. The observed streamflow data is partitioned into the model development (training) and model evaluation (testing) part, according to Fig. 2.

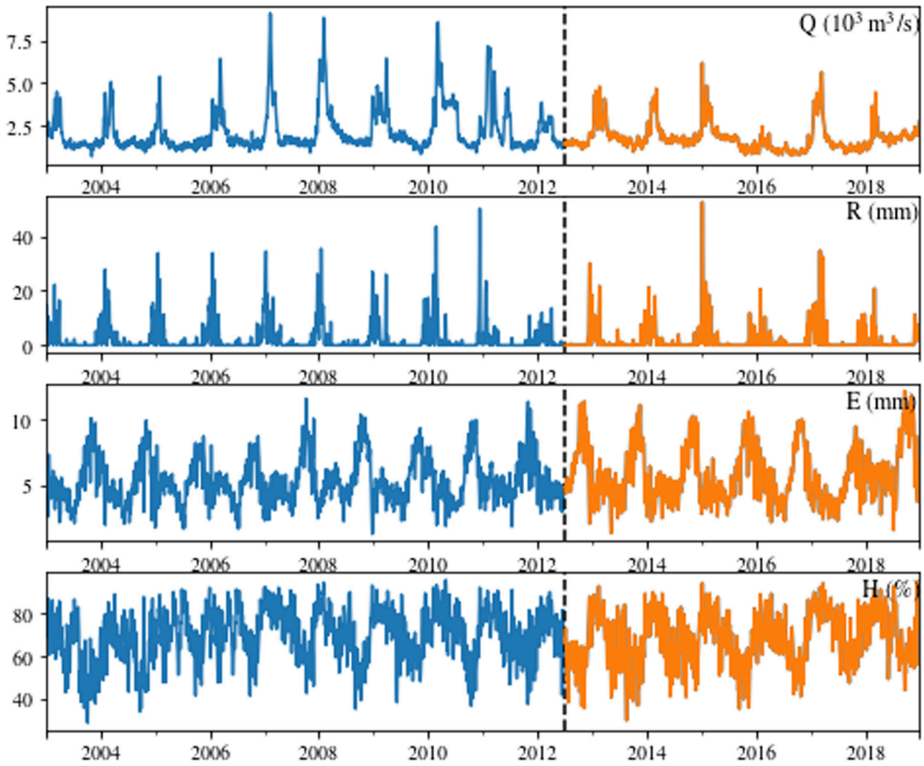

Fig. 2. Historical flow data of the Zambezi river at Cahora Bassa dam. The training set ranges from 2003-01-01 to 2012-06-30 and is shown in blue, while the test set ranges from 2012-07-01 to 2018-12-31 and appears in orange. (Color figure online)

2.2 Streamflow Estimation Model

To carry out the predictions, we selected periods with seven days in the historical flow series. The forecast model receives an input consisting of 28 values: 7 values of precipitation, 7 of evaporation, 7 humidity, and 7 values of flow. The model returns the corresponding estimated flow 7-days ahead. There are three forms of flow forecasting, with a few hours or days in advance, called short-term forecasts, medium-term or seasonal forecasts, and long-term forecasts, based on probabilities, trend analysis or climate change scenarios.

The estimated flow was considered as a function of finite sets of antecedent precipitation, evaporation and humidity and flow observations at the stations. The predictive model has the following form:

$$Q_{t+7} = F(\overbrace{R_t, \cdots, R_{t-6}}^{\text{rainfall}}, \underbrace{E_t, \cdots, E_{t-6}}_{\text{evaporation}}, \overbrace{H_t, \cdots, H_{t-6}}^{\text{humidity}}, \underbrace{Q_t, \cdots, Q_{t-6}}_{\text{streamflow}}) \quad (1)$$

where Q_{t+j} is the streamflow at day $t+j$, R_{t+j} is the precipitation (rainfall) at day $t+j$, E_{t+j} is the evaporation at day $t+j$, H_{t+j} is the humidity measured at day $t+j$, and F is an estimation function.

2.3 Extreme Learning Machine (ELM)

Extreme Learning Machine (ELM) [12] is a particular case of feedforward artificial neural network, where the vast majority is composed of only one hidden layer. Compared with the Artificial Neural Network (ANN), the Support Vector Machine (SVM), and other traditional prediction models, the ELM model retains the advantages of fast learning, good ability to generalize, and convenience in terms of modeling [7]. According to authors, these models are able to produce good generalization performance and learn thousands of times faster than networks trained using backpropagation. In literature, it also shows that these models can outperform support vector machines in both classification and regression applications.

In ELMs, there are three levels of randomness: (1) fully connected, hidden node parameters are randomly generated; (2) the connection between inputs to hidden nodes can be randomly generated, not all input nodes are connected to a particular hidden node; (3) a hidden node itself can be a subnetwork formed by several nodes resulting in learning local features.

The output function of ELM used in this paper is given by [18]

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{L} \beta_i G(\mathbf{w}_i, b_i, \mathbf{x}) \quad (2)$$

where $\hat{y}$ is the ELM prediction associated to the input vector $\mathbf{x}$, $\mathbf{w}_i$ is the weight vector of the ith hidden node, b_i are the biases of the neurons in the hidden layer, β_i are output weights, $G(\cdot)$ is the nonlinear activation function and L is the number of hidden nodes. The parameters $(\mathbf{w}, b)$ are randomly generated (normally distributed with zero mean and standard deviation equals to one), and weights β_i of the output layer are determined analytically. The activation functions $G(\mathbf{w}, b, \mathbf{x})$ with the hidden nodes weights $(\mathbf{w}, b)$ are shown in Table 1.

The output weight vector $[\beta_1, ..., \beta_L]$ can be determined by minimizing [11]

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^L} (\|\mathbf{H}\boldsymbol{\beta} - \mathbf{y}\| + C||\boldsymbol{\beta}||^2) \quad (3)$$

Table 1. Activation functions used in ELM. The hidden node parameters $(\mathbf{w}, b)$ are randomly generated using a normal distribution $N(0, 1)$.

#	Name	Activation function G
1	Identity	$G(\mathbf{w}, b, \mathbf{x}) = \|\|\mathbf{w} \cdot \mathbf{x} + b\|\|$
2	Sigmoid	$G(\mathbf{w}, b, \mathbf{x}) = \frac{1}{1+exp(-\mathbf{w}\cdot\mathbf{x}+b)}$
3	Hyperbolic Tangent	$G(\mathbf{w}, b, \mathbf{x}) = \frac{1-exp(\mathbf{w}\cdot\mathbf{x}+b)}{1+exp(\mathbf{w}\cdot\mathbf{x}+b)}$
4	Gaussian	$G(\mathbf{w}, b, \mathbf{x}) = exp(-(\mathbf{w} \cdot \mathbf{x} + b)^2)$
5	Multiquadrics	$G(\mathbf{w}, b, \mathbf{x}) = \sqrt{\|\|\mathbf{w} - \mathbf{x}\|\|^2 + b^2}$
6	Inverse Multiquadrics	$G(\mathbf{w}, b, \mathbf{x}) = 1/(\|\|\mathbf{w} - \mathbf{x}\|\|^2 + b^2)^{1/2}$
7	Swish	$G(\mathbf{w}, b, \mathbf{x}) = \frac{\|\|\mathbf{w}\cdot\mathbf{x}+b\|\|}{1+exp(-\mathbf{w}\cdot\mathbf{x}+b)}$
8	ReLU	$G(\mathbf{w}, b, \mathbf{x}) = \max_i (0, (\mathbf{w} \cdot \mathbf{x} + b))$

where $\mathbf{y}$ is the output data vector, $\mathbf{H}$ is the hidden layer output matrix

$$\mathbf{H} = \begin{bmatrix} G_1(\mathbf{w}_1, b_1, \mathbf{x}_1) & \cdots & G_L(\mathbf{w}_L, b_L, \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ G_1(\mathbf{w}_1, b_1, \mathbf{x}_N) & \cdots & G_L(\mathbf{w}_L, b_L, \mathbf{x}_N) \end{bmatrix} \text{ and } \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

is the output data vector with N the number of data points. The optimal solution is given by

$$\boldsymbol{\beta} = (\mathbf{H^T H})^{-1} \mathbf{H^T y} = \mathbf{H}^\dagger \mathbf{y}$$

where $\mathbf{H}^\dagger$ is the pseudoinverse of $\mathbf{H}$.

2.4 Parameter Tuning Guided by a Genetic Algorithm

We use in this work a Genetic Algorithm (GA) to find the internal parameters of the ELM neural network. In this scenario, each individual/candidate in the population represents an ELM neural network. There are practically four fundamental blocks for a genetic algorithm [10]: (i) Selection: the purpose of the selection is to choose the individuals who will serve as parents in the reproduction process. (ii) Crossover: this operator creates new individuals by mixing the characteristics of two parents. (iii) Mutation: this operator introduces diversity among the new individuals of the population. (iv) Reinsertion: elitism strategy is used to preserve part of the population that has superior performance. Hence, the best-known solutions found so-far in the search process were not lost. Using the these blocks we can describe GA with the following steps:

1. Create an initial population of random models (randomly generate a set of hyperparameters values);
2. Evaluate each individual (model) of the population and acquire their fitness value (performance metric of the model);
3. Select individuals for the recombination process;

4. Create a new population of new models (from a new set of internal parameters) generated through crossover and mutation on the selected individuals;
5. Combine the old population with the new one and keep only the best models (elitism strategy);
6. Repeat steps 2–5 until to satisfy the stopping criteria.

Table 2 shows the set of hyperparameters. Considering the ELM setup, a candidate solution $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$ represents the number of neurons in the hidden layers, the value of the parameter C in Eq. (3) and the activation function as shown in Table 1.

Table 2. Encoding of ELM candidate solutions. The column DV indicates the Decision Variable in the GA encoding.

DV	Description	Settings/Range
θ_1	No. neurons in the hidden layer, L	[1, 500]
θ_2	Regularization parameter C, Eq. (3)	[0.0001, 10000]
θ_3	Activation function G, Table 1	1: Identity; 4: Sigmoid; 3: Hyperbolic Tangent 4: Gaussian; 5: Multiquadric; 6: Inverse Multiquadric; 7: Swish; 8: ReLU;

3 Computational Experiments and Discussion

The internal parameters of each ELM model were set through the search performed by the Genetic Algorithm with a population size of 16 individuals evolving in 25 generations, crossover probability of 80%, and mutation rate of 10%. The fitness function is the RMSE calculated according to 3-fold cross-validation in the training set that ranges from 2003-01-01 to 2012-06-30. The training set appears in blue in Fig. 2. After the end of the evolutionary search, the best model's performance is calculated using the test set, a slice of the historical data ranging from 2012-07-01 to 2018-12-31 that appears in orange in Fig. 2. The lower and upper bounds $\boldsymbol{\theta}_L$ and $\boldsymbol{\theta}_U$, are given respectively by $\boldsymbol{\theta}_L = (1, 0.001, 1)$ and $\boldsymbol{\theta}_U = (100, 10^4, 8)$. Tournament selection was used to select individuals for the recombination process, using five individuals in the tournament. The experiments were repeated 25 times with different random seeds.

To assess the ELM performance, we have implemented three other machine learning methods in the same computational framework: ElasticNet linear model (EN) [9], Support Vector Regression (SVR) [4], and Extreme Gradient Boosting (XGB) [5]. All models had their internal parameters optimized by the genetic algorithm using the same population size, crossover probabilities, mutation rates, and the number of generations. The encondig for EN models involves three parameters, $(\theta_1, \theta_2, \theta_3)$, where θ_1 is a penalty term, θ_2 is the ratio between L_1 and L_2 regularization, and θ_3 is a bolean variavle that allows only positive coefficients. The upper and lower bounds are $\theta_1 \in [10^{-6}, 2]$, $\theta_2 \in [0, 1]$, and $\theta_3 \in \{\text{True}, \text{False}\}$. The SVR model implements RBF (radial basis function) kernel and each individual encodes three SVR parameters in the form

$(\theta_1, \theta_2, \theta_3) = (\gamma, C, \varepsilon)$. The lower and upper bounds are $\boldsymbol{\theta}_L = (0, 0.1, 0.01)$ and $\boldsymbol{\theta}_U = (1, 10^4, 100)$. The candidate solutions for XGB models encodes four parameters $(\theta_1, \theta_2, \theta_3, \theta_2)$, where θ_1 is the learning rate, θ_2 controls the number of estimators of the ensemble model, θ_3 is the maximum depth of each estimator, and θ_4 is the regularization parameter. The lower and upper bounds are $\boldsymbol{\theta}_L = (10^{-6}, 10, 1, 0)$ and $\boldsymbol{\theta}_U = (1, 100, 20, 100)$. Table 3 presents the metrics used in this paper and their brief description.

Table 3. Performance metrics. R^2 is the coefficient of determination, RMSE is the Root Mean Squared Error, while MAPE is the Mean Absolute Percentage Error. NSE is the Nash-Sutcliffe efficiency for the estimation model [17], and KGE is the Kling-Gupta efficiency between simulated and observed values [8]. O_i represents the observed data and P_i the predicted values. $\overline{O}$ is the mean observed streams. r is the Pearson product-moment correlation coefficient and α is the ratio between the standard deviation of the predicted values and the standard deviation of the observed values. Finally, β is the ratio between the mean of the predicted values and the mean of the observed values.

Metric acronym	Expression
R^2	$\frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}(O_i - \overline{O})^2}$
RMSE	$\frac{1}{N}\sqrt{\sum_{i=1}^{N}(O_i - P_i)^2}$
MAPE	$100 \times \frac{1}{N}\sum_{i=1}^{N}\frac{\lvert O_{(i)} - P_{(i)}\rvert}{\lvert O_{(i)}\rvert}$
NSE	$1 - \frac{\Sigma_{i=1}^{N}(O_i - P_i)^2}{\sum i{=}1^{n}(O_i - \overline{O})^2}$
KGE	$1 - \sqrt{(r-1)^2 + (\alpha - 1)^2 + (\beta - 1)^2}$

Table 4 presents the descriptive statistics for the performance metrics. The results produced by the ELM model are compared with other models of machine learning, such as EN, SVR, XGB. From this table, we can observe ELM produced competitive results concerning all metrics. However, ELM presents better estimates showing lower standard deviations.

Table 4. Averaged metrics produced by ELM and comparison with other approaches. The standard deviations appear within parentheses. The first column shows the metric acronym. The second column summarizes ELM results, the second and third columns, the results for EN and SVR, while the last column presents the metrics for XGB. A total of 25 runs were performed.

Estimator	ELM	EN	SVR	XGB
R^2	**0.71** (0.004)	0.66 (0.073)	0.71 (0.062)	0.70 (0.005)
RMSE	**0.42** (0.003)	0.47 (0.060)	0.42 (0.041)	0.43 (0.003)
MAPE	**14.85** (0.182)	16.61 (2.438)	15.20 (2.457)	16.05 (0.234)
NSE	**0.72** (0.003)	0.60 (0.118)	0.68 (0.025)	0.68 (0.005)
KGE	**0.86** (0.001)	0.81 (0.055)	0.82 (0.020)	0.84 (0.003)

The results of the statistical tests for all metrics are displayed in Table 5. The null hypothesis is that the mean in each evaluation metric is equal for all models. As can be seen, we reject the null hypothesis for MAPE, NSE, RMSE, and KGE because their p-values were smaller than the significance level of 0.05. This means that these metrics can be used as a criterion to evaluate the performance of the models in the forecast 7-days ahead. All models produced similar results for R^2, however, with relatively low. Table 5 shows the results of multiple comparisons applying the Tukey test ($\alpha = 0.05$) to ELM pairs and other models for each metric. The null hypothesis is that the means in each pair of models are equal, which leads to a similar conclusion in Table 6.

Table 5. *p*-values of ANOVA test for each metric.

Metric	R^2	RMSE	MAPE	NSE	KGE
p-value	0.101	0.032	0.020	0.000	0.000

Table 6. Pairwise Tukey test. The null hypothesis is that the estimators' means are equal. The entries show the outcome for rejecting the null hypothesis.

Estimator 1	Estimator 2	R^2	RMSE	MAPE	NSE	KGE
ELM	EN	False	True	False	True	True
ELM	SVR	False	False	False	True	True
ELM	XGB	False	False	True	True	True

Figure 3 shows the best hydrograph according to RMSE for the 7-days ahead flow Q_{t+7} in 25 runs. A hydrograph is a graph of the flow in a stream over a period in a specific location. From this figure, we observe that ELM is capable of representing the characteristics in the flow series, such as the change in level in the critical periods with lower flows and higher flows. The simulated hydrograph showed a very close behavior with good adherence to the observed data. However, the ELM solution overestimates the peak flows.

The internal parameters of ELM models produced by the genetic search were collected for all runs, and their distributions are shown in Fig. 4. This figure shows the parameter distribution for activation function G, the number of neurons in the hidden layer L, and the regularization parameter C. From this figure, we observe from a total of eight activation functions shown in Table 1, only three were chosen in the final solutions: Multiquadric, Swish, and Sigmoid. As shown in Fig. 4, the Multiquadric activation function was selected in 22 out of 25 runs, while the Swish function appears in the final solutions in two runs and the Sigmoid in one run. The number of neurons (parameter L) in the hidden layer is around 450, ranging from 350 to 500 neurons. More than half the solutions were set with 425–475 neurons, as shown in the boxplot's interquartile.

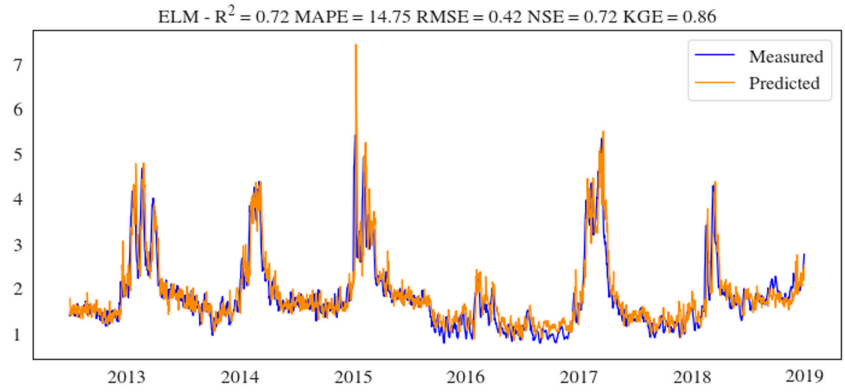

Fig. 3. Best solution according to RMSE for 7-day ahead streamflows.

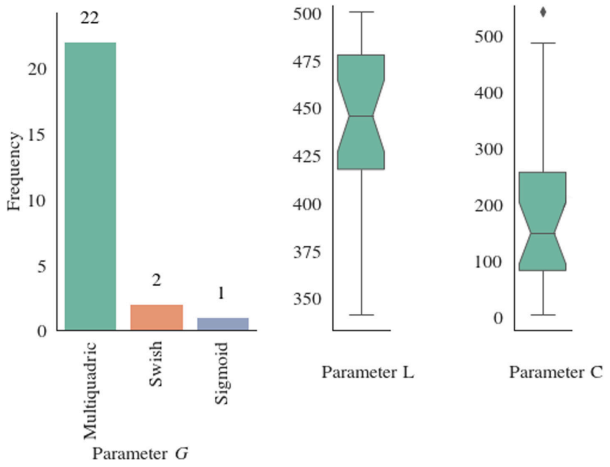

Fig. 4. Distribution of ELM parameters over 25 runs. The activation function G and the number of neurons L influence the solutions according to Eq. 2. The activation functions are described in Table 1. The regularization parameter C in Eq. 3 controls the smoothness.

The distribution of penalization shows the parameter C was set in a relatively wide range, but the boxplot shows more than 50% of the values lie in the interval [100–250].

Figure 5 displays the scatter plot of the solution with smallest RMSE over 25 runs. We observe that ELM produced solutions with a good agreement up to flows of 2500 m^3/s, indicated by the dotted line. Furthermore, it can be seen that the quality of the solutions deteriorates as the flow value of 7-days ahead increases. One can observe that the estimation of extreme events or extreme flows is difficult to predict by ELM. We highlight that accurately identifying extreme flows are critical to the decision policy of the dam operations. Due to the river

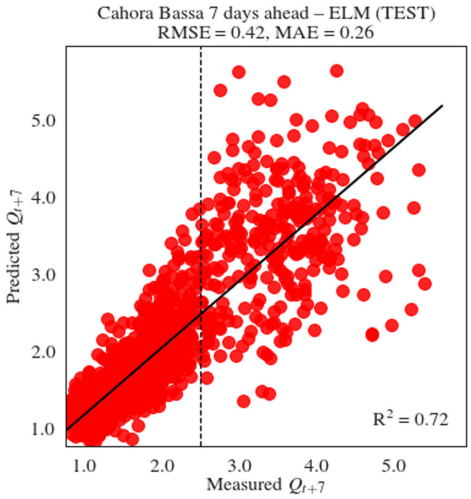

Fig. 5. Scatter plot of the solution with smallest RMSE over 25 runs.

bed's geographic characteristics downstream, the dam, the valley broadens, and the river develops a narrow floodplain [16]. As a result, extreme streamflows can abruptly change the river flow, which may cause floods and disasters. Accurate extreme streamflow predictions help safely control the river flow, allowing fit the demands for energy generation.

4 Conclusions

The present study shows the ability of the ELM model for the daily forecast of the natural flow to the Cahora Bassa dam. The results show that using a genetic algorithm to guide the selection of ELM parameters provides superior results in the flow forecast 7-days ahead. The ELM technique showed good potential to perform flow predictions; however, it seemed to overestimate peak flows. As observed in this study, ELM has been shown to obtain better results than EN, XGB, and SVR. Further studies include exploring other models, such as deep learning and online machine learning models to evaluate their forecasting capacity. The proposed ELM model can be useful for flow forecasting, which is most important to reservoir operation. It also helps to allocate the dam's waiting volume and optimize the operating rules, balancing the energy generation while ensuring the reservoir's ecological flows.

Acknowledgment. The authors acknowledge the Cahora Bassa Hydroelectric Company for the essential support during this work and the financial support from CNPq (429639/2016-3), FAPEMIG (APQ-00334/18), and CAPES - Finance Code 001.

References

1. Adnan, R.M., Liang, Z., Heddam, S., Zounemat-Kermani, M., Kisi, O., Li, B.: Least square support vector machine and multivariate adaptive regression splines for streamflow prediction in mountainous basin using hydro-meteorological data as inputs. J. Hydrol. **586**, 124371 (2020)
2. Ahmed, N.K., Atiya, A.F., Gayar, N.E., El-Shishiny, H.: An empirical comparison of machine learning models for time series forecasting. Econom. Rev. **29**(5–6), 594–621 (2010)
3. Alvisi, S., Franchini, M.: Fuzzy neural networks for water level and discharge forecasting with uncertainty. Environ. Model. Softw. **26**(4), 523–537 (2011)
4. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 1–27 (2011)
5. Chen, T., Guestrin, C.: Xgboost. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 2016
6. Claesen, M., Moor, B.D.: Hyperparameter search in machine learning. CoRR abs/1502.02127 (2015)
7. Guo, P., Cheng, W., Wang, Y.: Hybrid evolutionary algorithm with extreme machine learning fitness function evaluation for two-stage capacitated facility location problems. Expert Syst. Appl. **71**, 57–68 (2017)
8. Gupta, H.V., Kling, H., Yilmaz, K.K., Martinez, G.F.: Decomposition of the mean squared error and NSE performance criteria: implications for improving hydrological modelling. J. Hydrol. **377**(1), 80–91 (2009)
9. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. SSS. Springer, New York (2009). https://doi.org/10.1007/978-0-387-84858-7
10. Holland, J.H.: Hidden Order: How Adaptation Builds Complexity. Helix books, Basic Books, New York (1995)
11. Huang, G., Huang, G.B., Song, S., You, K.: Trends in extreme learning machines: a review. Neural Netw. **61**(Supplement C), 32–48 (2015)
12. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: a new learning scheme of feedforward neural networks. In: Proceedings of the IEEE International Joint Conference on Neural Networks, vol. 2, pp. 985–990. IEEE (2004)
13. Hussain, D., Khan, A.A.: Machine learning techniques for monthly river flow forecasting of Hunza River, Pakistan. Earth Sci. Inf. **13**(3), 939–949 (2020). https://doi.org/10.1007/s12145-020-00450-z
14. Kuhn, M., Johnson, K.: Applied Predictive Modeling. Springer, New York (2013). https://doi.org/10.1007/978-1-4614-6849-3
15. Li, J., Wang, Z., Lai, C., Zhang, Z.: Tree-ring-width based streamflow reconstruction based on the random forest algorithm for the source region of the Yangtze River, China. CATENA **183**, 104216 (2019)
16. McCartney, M., Beilfuss, R.D., Rebelo, L.-M.: Zambezi River Basin. In: Finlayson, C.M., Milton, G.R., Prentice, R.C., Davidson, N.C. (eds.) The Wetland Book, pp. 1217–1232. Springer, Dordrecht (2018). https://doi.org/10.1007/978-94-007-4001-3_91
17. Nash, J., Sutcliffe, J.: River flow forecasting through conceptual models part I - a discussion of principles. J. Hydrol. **10**(3), 282–290 (1970)
18. Saporetti, C.M., Duarte, G.R., Fonseca, T.L., da Fonseca, L.G., Pereira, E.: Extreme learning machine combined with a differential evolution algorithm for lithology identification. RITA **25**(4), 43–56 (2018)

19. Sun, A.Y., Wang, D., Xu, X.: Monthly streamflow forecasting using Gaussian process regression. J. Hydrol. **511**, 72–81 (2014)
20. Thakur, B., Kalra, A., Ahmad, S., Lamb, K.W., Lakshmi, V.: Bringing statistical learning machines together for hydro-climatological predictions - case study for Sacramento San Joaquin River Basin, California. J. Hydrol. Reg. Stud. **27**, 100651 (2020)
21. Wang, W.C., Chau, K.W., Cheng, C.T., Qiu, L.: A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. J. Hydrol. **374**(3), 294–306 (2009)