

Article

Stability Assessment of Rubble Mound Breakwaters Using Extreme Learning Machine Models

Xianglong Wei ^{1,2} , Huaixiang Liu ², Xiaojian She ², Yongjun Lu ^{2,*}, Xingnian Liu ¹ and Siping Mo ²

¹ State Key Laboratory of Hydraulics and Mountain River Engineering, College of Water Resource and Hydropower, Sichuan University, Chengdu 610065, China

² State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Nanjing Hydraulic Research Institute, Nanjing 210029, China

* Correspondence: yjlu@nhri.cn

Received: 7 August 2019; Accepted: 4 September 2019; Published: 7 September 2019



Abstract: The stability number of a breakwater can determine the armor unit's weight, which is an important parameter in the breakwater design process. In this paper, a novel and simple machine learning approach is proposed to evaluate the stability of rubble-mound breakwaters by using Extreme Learning Machine (ELM) models. The data-driven stability assessment models were built based on a small size of training samples with a simple establishment procedure. By comparing them with other approaches, the simulation results showed that the proposed models had good assessment performances. The least user intervention and the good generalization ability could be seen as the advantages of using the stability assessment models.

Keywords: breakwater; extreme learning machine; stability assessment; machine learning

1. Introduction

The determination of the armor unit's weight is an important component in the rubble mound breakwater design process, and the armor units play a key role in maintaining the stability of breakwaters under wave attacks. The armor unit's weight can be computed using the stability number of rubble mound breakwaters, N_s , which is commonly obtained from the Hudson formula [1] or the van der Meer formula [2]. Hudson's formula is widely used in breakwater design because of its convenient calculation of the unit mass and stability number. Some important physical factors are not involved in the formula, such as the wave period, wave length and the water depth in front of the breakwater, so other researchers did further research. van der Meer [2] proposed the following formula based on more than 300 runs of breakwater experiments under irregular wave attack. Compared to the Hudson formula, more parameters are included in the van der Meer formula (VM formula), such as the number of wave attacks and the breakwater permeability.

$$N_s = 6.2P^{0.18} \left(\frac{S_d}{\sqrt{N_w}} \right)^{0.2} \frac{1}{\sqrt{\xi_m}} \xi_m < \xi_c \quad (1)$$

$$N_s = 1.0P^{-0.13} \left(\frac{S_d}{\sqrt{N_w}} \right)^{0.2} \cot \alpha \xi_m^p \xi_m \geq \xi_c \quad (2)$$

where P is the permeability of the breakwater, S_d is the damage level, N_w is the number of wave attacks, ξ_m is the surf similarity parameter, ξ_c is the transition condition of the surf similarity, and α is the slope angle. N_s can be expressed as follows in the study of van der Meer [2]:

$$N_s = \frac{H_s}{\Delta D_n} \tag{3}$$

where H_s is the significant wave height, D_n is the nominal median diameter of the stones used in the breakwater, and Δ is the relative mass density, which can be expressed as follows:

$$\Delta = \frac{\rho_\gamma}{\rho} \tag{4}$$

where ρ_γ is the stone mass density and ρ is the water mass density.

Over the past decades, van der Meer’s formula has proven to be the most widely used formula to guide rubble mound breakwater design, while at the same time a developed formula was proposed by van der Meer [3,4] to assess the stability of the toe structure of rubble mound breakwaters, based on a new series of experiments. In addition to the studies of Hudson and van der Meer, some other researchers have proposed some novel formulas to assess the stability by analyzing the experimental results, such as [5], Hanzawa, et al. [6], Vidal, et al. [7], Etemad-Shahidi and Bonakdar [8] and Van Gent and Der Werf [9]. Previous studies have shown that the definition equation of the stability number N_s is stable, and that it is a function of the wave height and the relative density and nominal diameter of the stone. Meanwhile, in different studies, the damage condition of the breakwater section used to compute the stability number has various definitions. In the studies of Thompson [10] and Hanzawa [6], a damage parameter was proposed to describe the damage condition of the breakwater section, which was a function of the stone density, stone size, wave height, wave number and erosion area in a cross section, while in the studies of van der Meer [2,3] and Kajima [5], a simple damage level was proposed. A summary of the definitions is listed in Table 1 [11].

Table 1. Various definitions for the damage parameter (damage level).

Definition	Formula	Researcher
Damage parameter	$N_\Delta = \frac{A\rho_r 9D_{50}}{\rho_a D_{50}^3 \frac{\pi}{6}}$	Thompson and Shuttler (1975) [10]
Damage parameter	$N_0 = \left(\frac{H_{1/3}/\Delta D_n - 1.33}{2.32} \right)^2 N_w^{0.5}$	Hanzawa et al. (1996) [6]
Damage level	$S = \frac{A}{D_{50}^3}$	van der Meer (1988) [2]
Damage level	$N_{od} = \frac{M}{B/D_n}$	van der Meer (1998) [3]
Damage level	$S = 0.6D'$	Kajima (1994) [5]

Note: A is the erosion area in a cross section, ρ_r is the bulk density of the material as laid on the slope, ρ_a is the mass density of the stone, D_{50} is the median diameter of the stone, $H_{1/3}$ is the average wave height of the $N_w/3$ highest waves reaching a rubble mound breakwater of a sea state composed of N_w waves, M is the number of stones removed from the structure in a strip, B is the length of the test section, and D' is the ratio of the number of displaced units to the total number of units.

Besides the empirical formula, N_s can also be predicted by using the machine learning approaches. In the past two decades, a large and growing body of literature has investigated the machine learning approaches to assess the stability of rubble mound breakwaters, such as Artificial Neural Networks (ANN) [12–15], Fuzzy Neural Networks (FNN) [16,17], Model Tress (MT) [8,18], Support Vector Machine (SVM) [19–22], and Genetic Programing (GP) [23]. These studies have shown that the performance of machine learning approaches is better than that of the traditional formulas [23,24]. The study of Balas, Koç and Tür [13] provides new insights into improving the prediction accuracy of ANN models via the principal component analysis, which could reduce the needed amount of training data and transform the original data set into a set of uncorrelated variables that capture all of the variance of the original data set [25], but many methods still suffered from complex establishment

procedures and large demands for the training sets' size. Thus, reducing the parameter counts and the training data size, and simplifying the training process, should be a concern for further research.

Besides the empirical formula, N_s can also be predicted by using the machine learning approaches. In the past two decades, a large and growing body of literature has investigated the machine learning approaches to assess the stability of rubble mound breakwaters, such as Artificial Neural Networks (ANN) [12–15], Fuzzy Neural Networks (FNN) [16,17], Model Tress (MT) [8,18], Support Vector Machine (SVM) [19–22], and Genetic Programming (GP) [23]. These studies have shown that the performance of machine learning approaches is better than that of the traditional formulas [23,24]. The study of Balas, Koç and Tür [13] provides new insights into improving the prediction accuracy of ANN models via the principal component analysis, which could reduce the needed amount of training data and transform the original data set into a set of uncorrelated variables that capture all of the variance of the original data set [25], but many methods still suffered from complex establishment procedures and large demands for the training sets' size. Thus, reducing the parameter counts and the training data size, and simplifying the training process, should be a concern for further research.

The Extreme Learning Machine is a robust machine learning algorithm based on the Single-Hidden Layer Feedforward Network (SLFN) [26], which was very simple in t neural network architecture. Previous studies have shown that the ELM could be used in wide areas, such as classification [27–30] and regression [31–34], and that it showed a good generalization performance at fast learning speeds [35]. The main advantage of ELM is that the user-defined parameters for training an assessment ELM model only include the kind of activation function and the number of hidden neurons, which makes the model establishment very convenient; besides, the ELM model can obtain a high prediction accuracy based on a small size of training data sets. Based on this, the ELM method was proposed to develop a novel and simple stability assessment model of rubble mound breakwaters. This is the first study on the application of ELM in the stability assessment of rubble mound breakwaters. Therefore, the findings in this study make a contribution to the stability assessment and to the design of the rubble mound breakwaters.

This paper is organized as follows. The fundamentals of the ELM approach and the model establishment are introduced in Section 2. In Section 3, the application of the ELM approach for the stability assessment of rubble-mound breakwaters is discussed, and a comparison between the ELM approach and other approaches is given. The main findings of this paper are summarized in Section 4.

2. Extreme Learning Machine Models

The Extreme Learning Machine is widely used in regression and classification [36]. However, up to now, few studies involving applications of ELM in coastal engineering have been published. A search of the literature revealed that ELM was once used to predict the sea level, tide, and wave heights [37–40]. In the following subsection, a brief introduction about the fundamentals of Extreme Learning Machine models is given to clarify the process details of the ELM model establishment. More information about ELM models can be found in Huang, Zhu and Siew [26,35], Huang, Huang, Song and You [36].

2.1. Fundamental of Extreme Learning Machine Model

The goal of the learning process is to find the relation between input training data sets and output training labels. Considering an ELM neural network with n neurons in the input layer, l neurons in the hidden layer, and m neurons in the output layer, the general structure of ELM is shown in Figure 1:

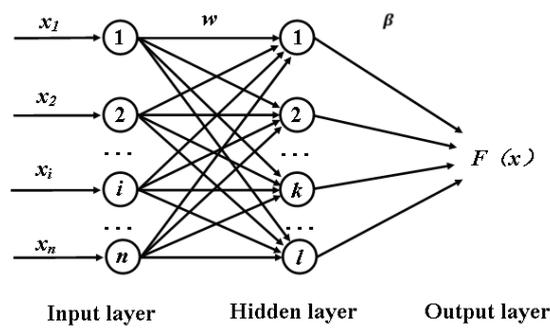


Figure 1. General structure of ELM.

The weight w between the neurons in the input layer and the neurons in the hidden layer can be expressed as:

$$w = \begin{bmatrix} \omega_{11} & \dots & \omega_{1n} \\ \vdots & \vdots & \vdots \\ \omega_{l1} & \dots & \omega_{ln} \end{bmatrix}_{l \times n} \tag{5}$$

where w_{ji} is the weight between the neuron i in the input layer and the neuron j in the hidden layer.

Meanwhile, the weight β between the neurons in the hidden layer and the neurons in the output layer can be expressed as:

$$\beta = \begin{bmatrix} \beta_{11} & \dots & \beta_{1m} \\ \vdots & \vdots & \vdots \\ \beta_{l1} & \dots & \beta_{lm} \end{bmatrix}_{l \times m} \tag{6}$$

where β_{jm} is the weight between the neuron j in the hidden layer and the neuron m in the output layer.

The bias b in the hidden layer is:

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_l \end{bmatrix} \tag{7}$$

For the given training samples X and the output matrix Y :

$$X = \begin{bmatrix} x_{11} & \dots & x_{1Q} \\ \vdots & \vdots & \vdots \\ x_{n1} & \dots & x_{nQ} \end{bmatrix}_{n \times Q} \tag{8}$$

$$Y = \begin{bmatrix} y_{11} & \dots & y_{1Q} \\ \vdots & \vdots & \vdots \\ y_{m1} & \dots & y_{mQ} \end{bmatrix}_{m \times Q} \tag{9}$$

Assuming that the activation function in the hidden layer was $g(x)$, then the output T is:

$$T = [t_1, t_2, \dots, t_Q], t_j = \begin{bmatrix} t_{1j} \\ \vdots \\ t_{mj} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^l \beta_{j1} g(w_i x_j + b_i) \\ \vdots \\ \sum_{i=1}^l \beta_{jm} g(w_i x_j + b_i) \end{bmatrix} (j = 1, 2, 3, \dots, Q) \tag{10}$$

where $\omega_i = [\omega_{i1}, \omega_{i2}, \dots, \omega_{in}]$, and $x_j = [x_{1j}, x_{2j}, \dots, x_{nj}]^T$.

Equation 11 can be rewritten in the following form:

$$H\beta = T' \tag{11}$$

where T' is the transposed matrix of T . H is the hidden layer output matrix of the neural network, which is as follows:

$$H(\omega_1, \omega_2, \dots, \omega_l, b_1, b_2, \dots, b_l, x_1, x_2, \dots, x_Q) = \begin{bmatrix} g(\omega_1, b_1, x_1) & \dots & g(\omega_l, b_l, x_1) \\ \vdots & & \vdots \\ g(\omega_1, b_1, x_Q) & \dots & g(\omega_l, b_l, x_Q) \end{bmatrix}_{Q \times l} \tag{12}$$

The minimum norm least-squares solution of $\min_{\beta} H\beta - T'$ is unique:

$$\hat{\beta} = H^+T' \tag{13}$$

where H^+ is the Moore-Penrose generalized inverse of the matrix of H .

More details about the ELM theory can be found in the studies of Huang, Zhu and Siew [35], Huang, Huang, Song and You [36].

The activation functions used in this paper are as follows:

Sigmoid function:

$$g(w_i x + b_i) = \frac{1}{1 + (\exp(-w_i x + b_i))} \tag{14}$$

Sin function:

$$g(w_i x + b_i) = \sin(w_i x + b_i) \tag{15}$$

Hardlim function:

$$g(w_i x + b_i) = \begin{cases} 1 & w_i x + b_i \geq 0 \\ 2 & w_i x + b_i < 0 \end{cases} \tag{16}$$

Trigonometric basis function:

$$g(w_i x + b_i) = \begin{cases} 1 - |w_i x + b_i| & -1 \leq w_i x + b_i \leq 1 \\ 0 & \text{else} \end{cases} \tag{17}$$

Radial basis function:

$$g(w_i x + b_i) = \exp\left(-\frac{\|x - w_i\|^2}{b_i^2}\right) \tag{18}$$

2.2. Model Establishment

Based on the damage level of the breakwater sections, two ELM models were built to predict the stability number of breakwaters. The first model (M1) was aimed at predicting the stability number of the breakwater sections whose damage level was in the range of 2 to 8. The other model (M2) was aimed at predicting the stability number of the breakwater sections whose damage level was in the range of 8 to 32 (8 was not included). High damage levels ($S > 8$) are not common in design practice [18]. Basheer and Hajmeer [41] (cited in Balas, Koç and Tür [13]) pointed out that there are no mathematical rules to determine the required amounts of training data and testing data. The number of the training data sets ranges from 90 to 579 in the previous studies discussed earlier. In the current study, 100 data points for training and 100 data points for testing were randomly selected for each model (M1, M2) from the experimental data of van der Meer [2], according to the damage level of the data, which can be found in in the Supplementary Materials. Five parameters we selected as the input nodes: the permeability, damage level, wave attack number, slope angle, and surf similarity parameter. The range of parameters used in the M1 and M2 models were presented in Figure 2 and Table 2.

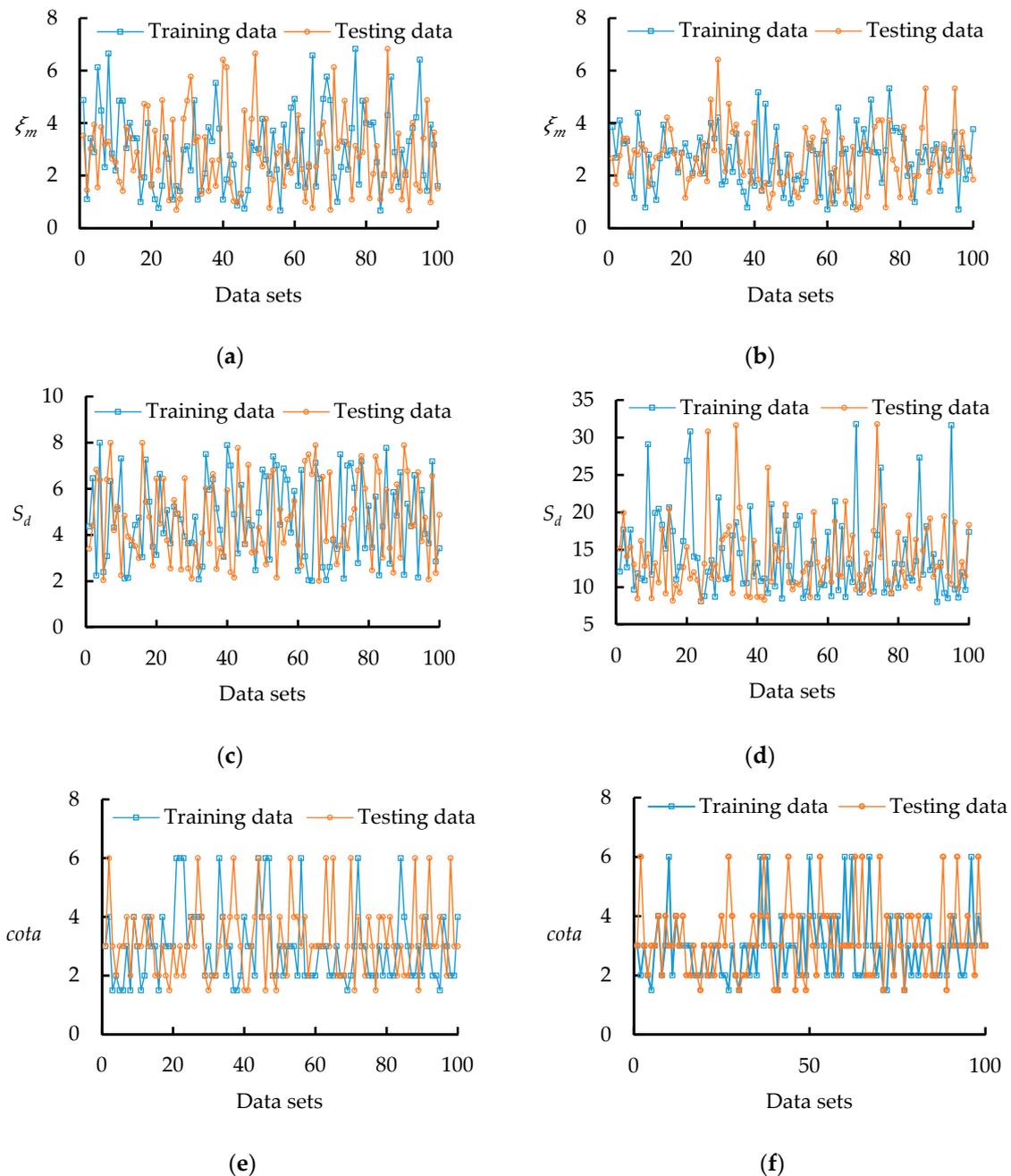


Figure 2. Parameters used in the M1 and M2 models: (a) the permeability of M1; (b) permeability of M2; (c) damage level of M1; (d) damage level of M2; (e) slope angle of M1; and (f) slope angle of M2.

Table 2. The range of parameters used in the M1 and M2 models.

Parameters	M1 Training Data	M1 Testing Data	M2 Training Data	M2 Testing Data
P	0.1, 0.5, 0.6	0.1, 0.5, 0.6	0.1, 0.5, 0.6	0.1, 0.5, 0.6
S_d	2–8	2–8	8–32	8–32
$cot a$	1.5–6	1.5–6	1.5–6	1.5–6
N_w	1000, 3000	1000, 3000	1000, 3000	1000, 3000
ξ_m	0.67–6.83	0.67–6.83	0.7–5.8	0.7–6.4
N_s	1.19–3.61	1.17–4.62	1.41–4.3	1.41–4.3

The flow chart of the ELM model establishment is shown in Figure 3. For each model, 5000 learning runs from the training data are necessary before the assessment model establishment. In order to evaluate the assessment performance of these ELM models, the bias (*BIAS*), correlation coefficient (*CC*), scatter index (*SI*) and index of agreement (*I_a*) are introduced as follows:

$$BIAS = \sum_{i=1}^N \frac{1}{N} (Y_i - X_i) \tag{19}$$

$$SI = \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - X_i)^2}}{\bar{X}_i} \tag{20}$$

$$CC = \frac{\sum_{i=1}^N (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^N (X_i - \bar{X})^2 \sum_{i=1}^N (Y_i - \bar{Y})^2}} \tag{21}$$

$$I_a = 1 - \frac{\sum_{i=1}^N (Y_i - X_i)^2}{\sum_{i=1}^N (|(Y_i - \bar{X})| + |(X_i - \bar{X})|)^2} \tag{22}$$

where X_i are the measured values, and their average is \bar{X} ; Y_i are the predicted values, and their average is \bar{Y} ; and N is the number of observations.

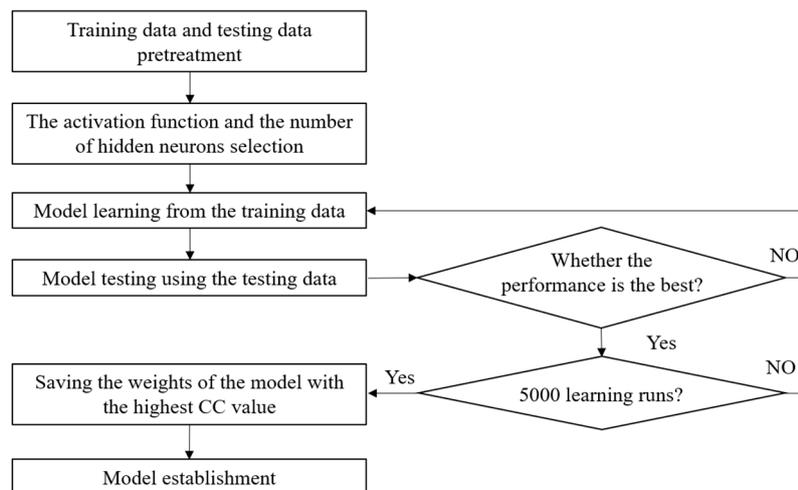


Figure 3. Flow chart of the ELM model establishment.

A search of the literatures shows that the *CC* parameter was the most widely used index to evaluate the model performances, meanwhile, when the *CC* value of the model reaches the highest value, the *I_a* value may not, so in the training process, the value of *CC* was selected as the only evaluation index of the models. After the training process, the weights of the model with the highest *CC* value are recorded, which can be used for the stability assessment. The data used for establishing the models is provided in the Supplementary Materials. The weights used in the M1 model and the M2 model are listed in Appendix A. The code used in this paper is provided in Appendix B.

3. Results and Discussion

3.1. The Influence of Hidden Neurons on the Assessment Performance of ELM Models with Different Activation Functions

The assessment performances of the ELM models are mainly determined by the number of the hidden neurons and the activation functions. Low damage levels ($S < 8$) are more common in the design practice for rubble mound breakwaters, as previously mentioned [18], and the selection of training data has little influence on this issue, so the M1 models ($S < 8$) were built to research the influence of the hidden neurons and the activation functions on the model performance.

Figure 4 shows the assessment performance of ELM models with different activation functions and different numbers of hidden neurons. The number of hidden neurons of these models were set from 5 to 90 with an interval of 5. It can be seen from the figures that the evaluation criteria CC and I_a are increasing following the addition of hidden neurons when the number of the hidden neurons is within the range of 5 to 20; then, the values of CC and I_a stay relatively constant when the number of the hidden neurons is above 20. The assessment performance criteria have a rapidly decreasing trend as the hidden neurons number increases in the range of 50 to 90, which indicates that too many hidden neurons lead to over-fitting. The models with the best assessment performance were the models built with 40–50 hidden neurons in the hidden layer, no matter what activation function was used in the models. What stands out in Figure 4 is that, for an ELM model with a randomly selected number of hidden neurons from 20 to 50, I_a is no less than 0.9 and CC is no less than 0.85. The selection of the activation function has little influence on the best performance of each model, which is mainly because the training data has few noisy data sets and is pure enough. The simulation results show that the ELM algorithm has a good generalization performance for the stability assessment.

A comparison of these results reveals that the ELM model with the maximum CC and I_a was the model built with 45 hidden neurons and the Sin function. The parameters for the model training and application were determined for these conditions.

3.2. Predicted Performance Comparison of Different Methods

Based on the previous results, the M1 assessment model and the M2 assessment model were built using the ELM algorithm with 45 hidden neurons in the hidden layer and the Sin function as the activation function. The M1 model was used to assess the stability numbers of the breakwater sections at a low damage level ($2 \leq S \leq 8$), and the M2 model was used to assess the stability numbers of the breakwater sections at a high damage level ($8 < S < 32$). In order to clarify the advantages of the M1 and M2 models, the VM formula [2], the EB formula [8], which was built using the model trees models with a high prediction accuracy, and the GPM1 formula [23], were selected to assess the stability using the same testing data. In the literatures, the EB formula and the GP method have the highest I_a values among those machine learning approaches, which indicates the best prediction performance. The GPM3 formula was not obtained in [23], and the GPM1 model has a similar performance to the GPM3 model presented in [23], so the GPM1 model was selected. These assessment results, predicted by different methods, were prepared and are shown in Figures 5 and 6, respectively.

Figure 5 provides the assessment performance for the breakwater sections that were at a low damage level using the VM formula, EB formula, GPM1 formula and M1 model. As shown in Figure 5, more than half of the predicted stability numbers of the breakwater sections were smaller than the measured values using the VM formula, the EB formula and the GPM1 formula. The assessment performance of the M1 model was more balanced: about half of the predicted stability numbers were smaller than the measured values, while the other half were larger than the measured values.

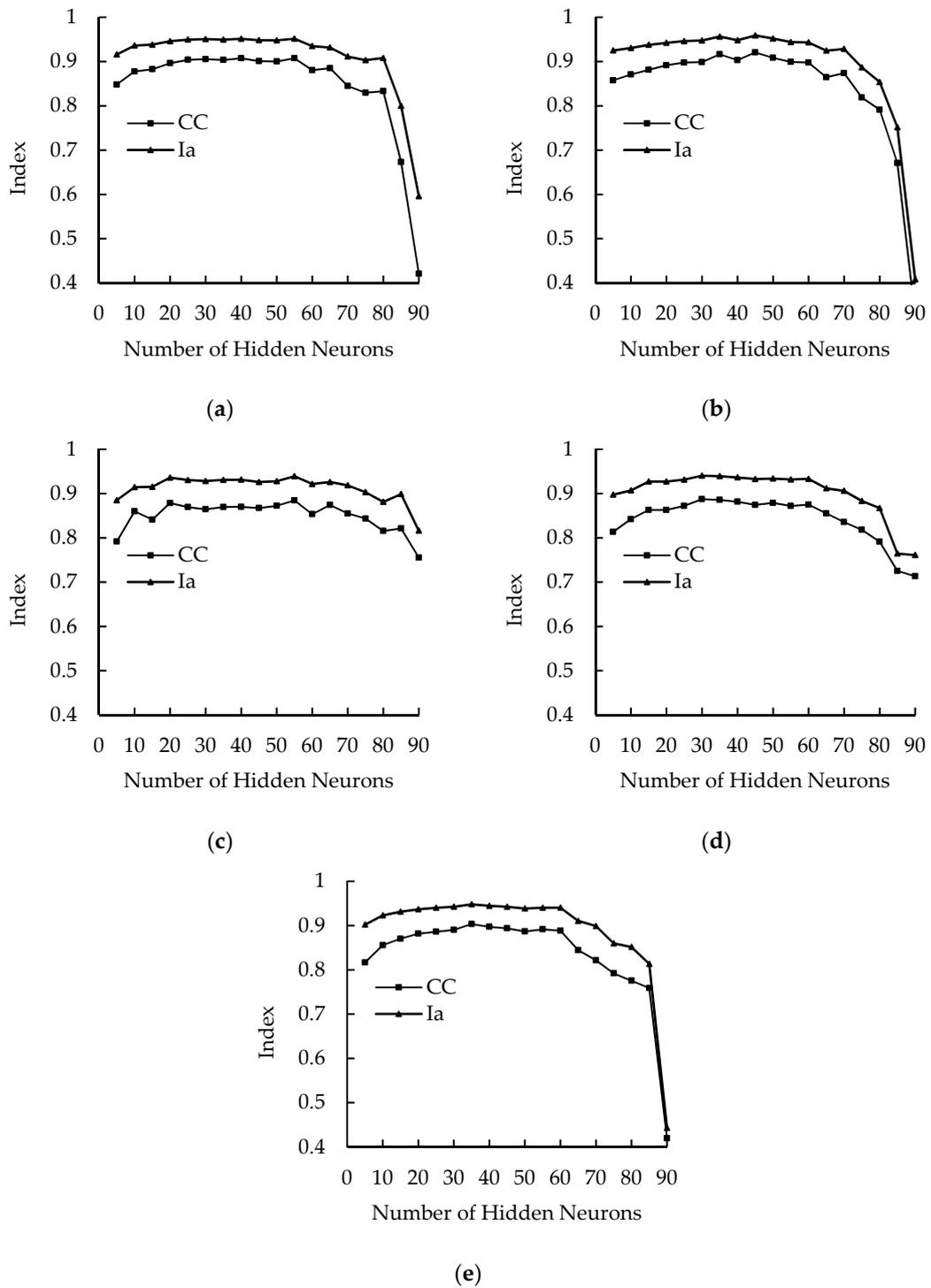


Figure 4. Performances of the ELM models with different hidden nodes: (a) Sigmoid function; (b) Sin function; (c) Hardlim function; (d) Tribas function; and (e) Radbas function.

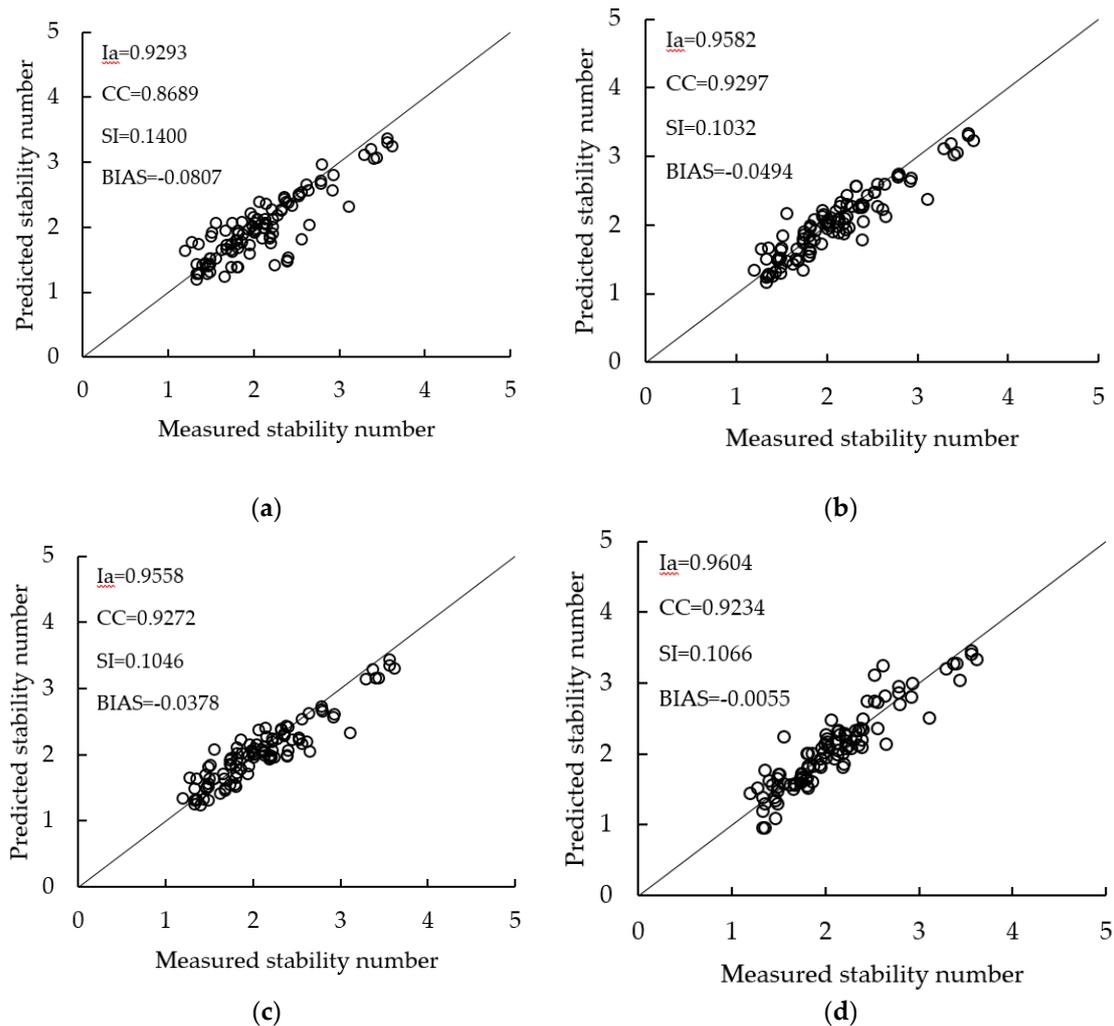


Figure 5. A performance comparison of different methods: (a) the van der Meer formula; (b) Etemad-Bonakdar formula; (c) GPM1 formula; and (d) M1 model.

Several statistical indices were introduced to assess the prediction performance of these approaches, such as the *BIAS*, *CC*, *SI*, and I_a . Lower values of *BIAS* and *SI* represent a better assessment performance, and higher values of *CC* and I_a indicate a better prediction agreement. When the values of *CC* and I_a are close to 1, this indicates a perfect agreement between the predicted and measured stability numbers. Table 3 lists the statistical index values of the three approaches. As shown in the table, the *CC* and I_a values of the VM formula are the smallest among these three assessment approaches, while the *BIAS* and the *SI* values are the largest, which indicates that the performance of the VM formula has the lowest quality agreement. The *CC* and I_a values of the EB formula, the GPM1 formula and the M1 model are nearly the same, while on the other hand the *SI* values of the two methods are also nearly the same. The evaluation indices show that the EB formula, the GPM1 formula and the M1 model have similar abilities for predicting the stability number of breakwaters with a low damage level, but that the M1 model was built based on a smaller size of training data, which indicates that the M1 model has a good generalization ability.

The performance of different approaches was also evaluated for a wider range of damage levels, i.e., $8 < S < 32$. The performances of the VM formula, EB formula, GPM1 formula and M2 model are discussed in the following paragraphs. All the predicted results are shown in Figure 6.

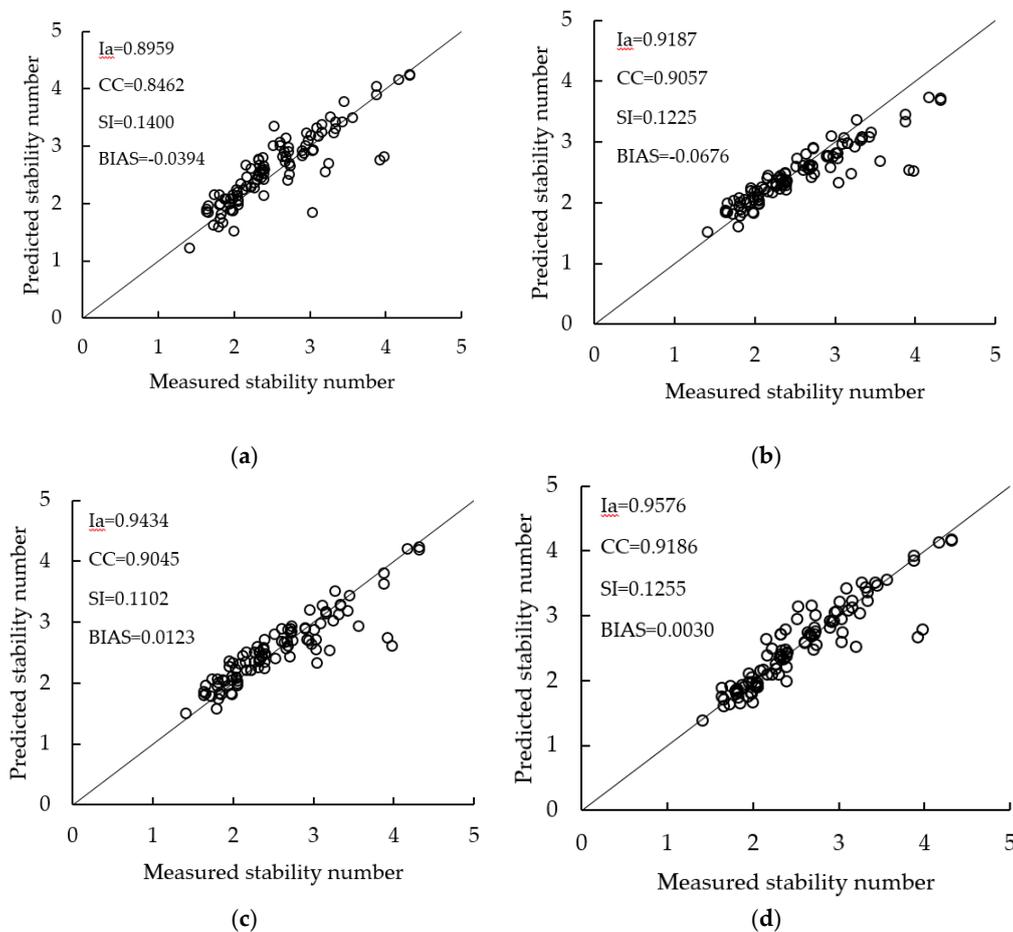


Figure 6. A performance comparison of different methods: (a) the van der Meer formula; (b) Etemad-Bonakdar formula; (c) GPM1 formula; and (d) M2 model.

Table 3. The evaluation indices of the performance of different models ($2 \leq S \leq 8$).

Methods	BIAS	SI	CC	I_a
VM	-0.0807	0.1400	0.8689	0.9293
EB	-0.0494	0.1032	0.9297	0.9582
GPM1	-0.0378	0.1046	0.9272	0.9558
ELM(M1)	-0.0055	0.1066	0.9234	0.9604

As shown in Figure 6, the VM formula overestimates most of the stability numbers; the same finding could also be found in the study of Etemad-Shahidi and Bali [18]. Meanwhile, on the opposite side, many stability numbers are underestimated by the EB formula. The number of overestimated and underestimated stability numbers is nearly the same for the predictions using the GPM1 model and the M2 model. In addition, the stability numbers predicted by the M2 model are more concentrated on the line of complete agreement than those of the other two approaches. The evaluation indices of these approaches are listed in Table 4. As seen in Table 4, the CC and I_a of the M2 model are the highest, while the BIAS and SI of the M2 model are the lowest, which indicates that the M2 model has the best performance for the stability number prediction of the breakwater sections with a wider damage level range.

Further research was done on the comparison of the assessment performance of different artificial neural network approaches from previous studies. The training parameters and the evaluation indices for these models are listed in Table 5. As can be seen, many kinds of machine learning approaches have been applied in predicting the stability numbers for breakwaters. The number of input parameters

of the training models have ranged from 4 to 8, and the number of the training data sets has ranged from 100 to 554. In the current paper, 5 input parameters and 100 sets of training data were used in the training processes of the M1 and M2 models, and the I_a values of the M1 and M2 models are higher than those in many of the previous studies, whose models were built using a larger size of training data and more input parameters. It should also be noted that the I_a or CC values should not be the only evaluation criteria in comparing different methods, since the testing data for each model was not the same. A comparison between the MT2 model (EB formula), GPM1 formula and the ELM method was made in the previous section by using the same testing data, which presented the advantages of the ELM method. The testing data for the MT2 model and the GPM1 formula in [8,23] is not the same testing data used in this paper, which leads to the different I_a and CC values presented in Tables 3–5. The CC values of HNN models in the study of Balas, Koç and Tür [13] are slightly higher than the CC values of the M1 and M2 models in the current paper, while the training data is pre-processed by using the principal component analysis (PCA), and the original data sets are 554 sets of experimental data. The PCA could remove the noisy data from the training data and extract the required information [13], so the use of PCA enhances the prediction ability of the machine learning models. It could also be expected that a PCA-ELM model will get a better prediction performance.

Table 4. The evaluation indices of the performance of different models ($8 < S < 32$).

Methods	BIAS	SI	CC	I_a
VM	-0.0394	0.1400	0.8462	0.8959
EB	-0.0676	0.1225	0.9057	0.9189
GPM1	0.0123	0.1102	0.9045	0.9434
ELM(M2)	0.0030	0.1022	0.9186	0.9576

Table 5. Calculation details of different machine learning approaches.

Researchers		CC	I_a	Training Data	Input Parameters	Testing Data
Mase, Sakamoto and Sakai [24]		0.91		100	6	No
Dong and Park [12]	I		0.914	100	6	641
	II		0.906	100	5	641
	III		0.902	100	6	641
	IV		0.915	100	7	641
	V		0.952	100	8	641
Kim, Dong and Chang [15]	I	0.905	0.948	207	5	119
	II	0.913	0.954	201	5	114
Erdik [16]	FL		0.945	579	6	579
Balas, Koç and Tür [13]	HNN-1	0.936		180 (PCA)	5	76
	HNN-2	0.927		180 (PCA)	4	76
Koç and Balas [17]	GA-FNN		0.932	166 (PCA)	5	42
	HGA-FNN		0.947	166 (PCA)	5	42
Etemad-Shahidi and Bonakdar [8]	MT1	0.931	0.97	386	5	193
	MT2	0.968	0.976	386	6	193
Koc, Balas and Koc [23]	GPM1		0.98	207	7	372
	GPM2		0.95	40	7	22
	GPM3		0.989	207	7	372
	GPM4		0.991	40	7	22
	VM		0.969			372
	VM		0.65			22
Current Study	ELM-M1	0.923	0.960	100	5	100
	ELM-M2	0.919	0.958	100	5	100

4. Conclusions

The aim of the present research was to develop novel and simple Extreme Learning Machine models to predict the stability number for rubble mound breakwater sections. Two ELM models were established to predict the stability number for breakwater sections: the M1 model for low damage levels ($2 \leq S \leq 8$) and the M2 model for high damage levels ($8 < S < 32$). It was shown that the prediction performances of ELM models were determined by the hidden neurons number, the size of the training data sets and the leaning times; furthermore, the selection of the activation function had little influence on the performance of these models. A comparison of the ELM models and other approaches suggested that the ELM models achieve a good performance with small user-defined parameters and a small training data set size. The key strength of this ELM approach is its good generalization ability and the simple process in establishing the models, which suggested that the ELM models could be an effective and simple tool for breakwater design and stability assessment. In the future, it would be interesting to study the hybrid ELM model, such as the PCA-ELM model, to assess the stability number for rubble mound breakwaters.

Supplementary Materials: The following are available online at <http://www.mdpi.com/2077-1312/7/9/312/s1>, training data of M1; testing data of M1; training data of M2; testing data of M2.

Author Contributions: Conceptualization, X.W. and Y.L.; methodology and software, X.W.; writing—original draft preparation, X.W., H.L.; writing—review and editing, Y.L., H.L., X.S., S.M. and X.L.; supervision, Y.L.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 51520105014), the Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. KYCX19_2328), and the National Key R&D Program of China (Grant No. 2016YFC0402108 & Grant No. 2017YFC0405900).

Acknowledgments: The authors wish to sincerely thank van der Meer for his free database on the stability of rubble mound breakwaters. We also appreciate the reviewers for their valuable suggestions and questions.

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

Appendix A

Table A1. Weights of M1.

0.5241	26.4975	0.8109	−0.4255	0.6935	−0.2008
0.5979	−3.6891	0.8853	0.6774	0.6278	−0.9448
0.6609	4.1169	0.1518	0.2682	0.2991	0.0216
0.9402	1.1036	−0.5524	0.7574	0.7449	−0.6862
0.1974	−32.1789	−0.3415	0.2084	−0.7634	−0.1137
0.8710	−3.2400	−0.6313	0.8539	−0.3247	0.9143
0.7430	−29.8057	0.3799	−0.2154	−0.7326	−0.9678
0.2418	5.5068	−0.8844	0.7903	0.7482	−0.7737
0.5977	−22.8297	0.1486	−0.9762	0.4362	0.0493
0.7125	1.7637	0.1595	0.5090	−0.9902	0.1901
0.1448	21.5258	−0.5829	0.0194	−0.7497	−0.6622
0.4441	−4.4733	−0.8343	0.7352	0.8387	−0.2093
0.1918	6.0145	−0.7073	0.0374	0.8523	0.7221
0.7374	52.4679	−0.4033	−0.0100	0.8501	−0.7359
0.1496	−9.1100	0.6958	−0.2846	0.9258	−0.7403
0.1726	−3.6265	0.4105	−0.9711	0.5676	−0.2996
0.8718	−8.6176	−0.9463	−0.0608	−0.5387	−0.6198
0.8638	32.5670	0.7170	0.0887	−0.4715	−0.5685
0.2632	13.7757	0.1976	−0.0730	−0.7203	−0.0630
0.1091	−9.6165	0.0072	−0.6522	0.2843	0.5996
0.3324	3.5370	0.3308	0.2424	−0.6404	0.6969

Table A1. Cont.

$BHN_1=$	0.1969	$lnW_1=$	-30.1128	$lnW_2=$	-0.7382	-0.5162	-0.7029	0.4078
	0.5033		40.9492		-0.1724	-0.1571	0.2812	-0.6081
	0.7217		-2.1417		0.0027	-0.1367	0.9792	-0.1936
	0.0935		-4.5602		0.7380	-0.4168	0.7734	-0.7967
	0.8949		-7.4840		-0.8876	-0.7521	0.7573	0.1826
	0.9296		-51.8195		-0.3970	0.0788	0.6631	-0.9412
	0.3114		-32.1941		0.5991	0.3968	-0.0596	0.2747
	0.8365		0.7267		0.9239	0.6791	0.7207	-0.1689
	0.6055		35.3792		-0.4155	-0.4794	-0.8263	-0.0045
	0.1465		5.8143		-0.9828	-0.4143	0.2699	0.9241
	0.9326		-14.1016		0.5911	0.8271	0.5772	0.0635
	0.1928		9.7569		-0.4223	-0.3700	0.2338	0.7443
	0.4138		2.4507		-0.1683	-0.2665	-0.5608	0.6952
	0.0855		-7.5543		-0.9139	-0.9217	-0.7361	-0.3699
	0.7125		8.2359		-0.7147	0.3655	-0.7379	-0.7774
	0.5891		3.9906		0.4442	0.7030	0.2163	0.0113
	0.8273		1.0311		0.9852	0.9763	-0.4108	-0.4178
	0.4677		11.6137		-0.2928	-0.8980	-0.1545	0.3437
	0.6765		6.5585		0.2751	0.9346	0.7867	0.6949
	0.3229		-7.2543		-0.1302	0.1766	0.9851	-0.9479
	0.7244		-8.8476		-0.4926	0.8206	0.0350	-0.9965
	0.1206		-10.8684		0.0382	-0.5207	0.0727	0.9225
	0.5268		-5.1499		-0.1425	-0.2191	0.4494	-0.9388
	0.2891		8.4672		0.6724	0.1706	-0.4620	0.9983

Table A2. Weights of M2.

0.4319	-42.0761	-0.9071	-0.2011	0.4272	0.8116
0.0320	-20.9949	-0.3291	-0.0591	-0.4020	0.8705
0.5944	-32.7021	-0.8404	-0.6064	0.8841	0.6630
0.6627	43.8901	-0.7591	-0.2472	0.8186	0.9823
0.9264	13.2387	0.8394	-0.8762	-0.1618	-0.3568
0.5949	-20.2892	0.5871	0.8688	-0.0913	0.7016
0.8525	-63.1985	0.3422	-0.7897	-0.4640	-0.2132
0.8806	-116.6938	0.2035	-0.5851	-0.2849	-0.8588
0.6270	-13.5202	0.7838	0.9148	-0.8121	0.6147
0.2328	35.2013	-0.1258	-0.3481	-0.7869	-0.1297
0.2941	3.1475	-0.8012	0.0277	-0.4674	-0.6218
0.2577	12.4026	-0.8559	-0.6591	-0.9608	0.2650
0.6162	0.2601	-0.4507	-0.2077	-0.4970	0.7523
0.1584	-3.0225	0.9716	0.8243	-0.4446	0.6805
0.5654	-7.7516	-0.6291	-0.5789	-0.5272	0.3921
0.5730	5.6515	-0.2855	-0.5305	0.4384	0.8396
0.6728	27.5190	0.0217	0.4931	-0.1090	-0.5729
0.7424	0.7453	-0.4198	0.1380	0.4327	-0.9063
0.7593	36.1610	0.4848	0.2726	0.6648	0.6994
0.7122	44.1426	-0.6639	-0.4197	0.5753	0.2885
0.6100	-16.2722	-0.0565	-0.0394	0.8366	-0.8595

Table A2. Cont.

$BHN_2=$	0.0537	$InW_2=$	-1.1557	$InW_2=$	-0.7270	-0.1948	-0.1188	-0.1234
	0.4458		22.3288		0.5387	0.8696	-0.4597	0.9628
	0.8475		-1.6268		0.7513	-0.9025	-0.6607	0.6460
	0.9733		-83.7627		0.3622	-0.6518	-0.4596	-0.5125
	0.8544		22.4302		0.8799	-0.2234	-0.4083	-0.7212
	0.3858		-21.6072		-0.5399	0.1999	0.1068	-0.4392
	0.9096		-2.8837		-0.4029	-0.6285	0.8447	-0.2820
	0.1069		-28.1949		-0.7637	0.7851	-0.3326	-0.1881
	0.2582		-18.8255		-0.0014	-0.1194	0.5881	0.2810
	0.5765		47.8772		0.5480	-0.4075	-0.7139	-0.6196
	0.3990		-5.0568		0.8476	0.1595	-0.4691	0.1434
	0.3779		9.6864		0.7929	-0.4492	-0.8683	-0.4401
	0.3411		10.6359		-0.4233	0.5854	-0.9226	0.3489
	0.2897		3.4956		0.8980	-0.7244	0.0454	0.2533
	0.7287		45.6903		-0.4528	0.5858	0.1254	-0.0241
	0.7738		-45.6212		0.8116	-0.2095	-0.0985	-0.6733
	0.5252		25.9720		0.2493	-0.7998	-0.2112	-0.3585
	0.8545		-50.5003		-0.0441	0.5296	-0.0523	0.2865
	0.0416		3.4476		-0.8948	0.9645	-0.8378	0.9041
	0.6695		8.8539		-0.6859	-0.9783	-0.8757	-0.9541
	0.8819		20.7560		0.0010	0.8347	-0.0483	-0.2737
	0.9352		133.8480		0.2844	-0.8046	-0.2266	-0.8481
	0.1300		5.0666		0.6626	-0.6074	0.4537	-0.5816
	0.9134		14.3699		-0.5942	0.5127	-0.0012	-0.3285

Appendix B

The code for the Extreme Learning Machine can be downloaded on the following website: <http://www.ntu.edu.sg/home/egbhuang/>.

References

- Hudson, Y. Laboratory investigation of rubble-mound breakwater. *Proc. ASCE* **1959**, *85*, 93–122.
- Van der Meer, J.W. *Rock Slopes and Gravel Beaches under Wave Attack*; Delft University of Technology: Delft, The Netherlands, 1988.
- Pilarczyk, K. *Dikes and Revetments Design, Maintenance and Safety Assessment*; Routledge: London, UK, 1998.
- Meulen, T.v.d.; Schiereck, G.J.; d'Angremond, K. Rock toe stability of rubble mound breakwaters. *Coast. Eng.* **1996**, *83*, 1971–1984.
- Kajima, R. A new method of structurally resistive design of very important seawalls against wave action. In Proceedings of the Wave Barriers in Deepwaters, Yokosuka, Japan, 10–14 January 1994; pp. 518–536.
- Hanzawa, M.; Sato, H.; Takahashi, S.; Shimosako, K.; Takayama, T.; Tanimoto, K. Chapter 130 New Stability Formula for Wave-Dissipating Concrete Blocks Covering Horizontally Composite Breakwaters. In Proceedings of the Coastal Engineering 1996, Orlando, FL, USA, 2–6 September 1996.
- Vidal, C.; Medina, R.; Lomónaco, P. Wave height parameter for damage description of rubble-mound breakwaters. *Coast. Eng.* **2006**, *53*, 711–722. [[CrossRef](#)]
- Etemad-Shahidi, A.; Bonakdar, L. Design of rubble-mound breakwaters using M5' machine learning method. *Appl. Ocean Res.* **2009**, *31*, 197–201. [[CrossRef](#)]
- Van Gent, M.R.A.; Der Werf, I.V. Rock toe stability of rubble mound breakwaters. *Coast. Eng.* **2014**, *83*, 166–176. [[CrossRef](#)]
- Thompson, D.M.; Shuttler, R.M. *Riprap Design for Wind-Wave Attack, a Laboratory Study in Random Waves*; HR Wallingford: Wallingford, UK, 1975.
- Wei, X.; Lu, Y.; Wang, Z.; Liu, X.; Mo, S. A Machine Learning Approach to Evaluating the Damage Level of Tooth-Shape Spur Dikes. *Water* **2018**, *10*, 1680. [[CrossRef](#)]
- Dong, H.K.; Park, W.S. Neural network for design and reliability analysis of rubble mound breakwaters. *Ocean Eng.* **2005**, *32*, 1332–1349.

13. Balas, C.E.; Koç, M.L.; Tür, R. Artificial neural networks based on principal component analysis, fuzzy systems and fuzzy neural networks for preliminary design of rubble mound breakwaters. *Appl. Ocean Res.* **2010**, *32*, 425–433. [[CrossRef](#)]
14. Dong, H.K.; Kim, Y.J.; Dong, S.H. Artificial neural network based breakwater damage estimation considering tidal level variation. *Ocean Eng.* **2014**, *87*, 185–190.
15. Kim, D.; Dong, H.K.; Chang, S. Application of probabilistic neural network to design breakwater armor blocks. *Ocean Eng.* **2008**, *35*, 294–300. [[CrossRef](#)]
16. Erdik, T. Fuzzy logic approach to conventional rubble mound structures design. *Expert Syst. Appl.* **2009**, *36*, 4162–4170. [[CrossRef](#)]
17. Koç, M.L.; Balas, C.E. Genetic algorithms based logic-driven fuzzy neural networks for stability assessment of rubble-mound breakwaters. *Appl. Ocean Res.* **2012**, *37*, 211–219. [[CrossRef](#)]
18. Etemad-Shahidi, A.; Bali, M. Stability of rubble-mound breakwater using H 50 wave height parameter. *Coast. Eng.* **2012**, *59*, 38–45. [[CrossRef](#)]
19. Kim, D.; Dong, H.K.; Chang, S.; Lee, J.J.; Lee, D.H. Stability number prediction for breakwater armor blocks using Support Vector Regression. *KSCE J. Civ. Eng.* **2011**, *15*, 225–230. [[CrossRef](#)]
20. Narayana, H.; Loksha; Mandal, S.; Rao, S.; Patil, S.G. Damage level prediction of non-reshaped berm breakwater using genetic algorithm tuned support vector machine. In Proceedings of the Fifth Indian National Conference on Harbour and Ocean Engineering (INCHOE2014), Goa, India, 5–7 February 2014.
21. Harish, N.; Mandal, S.; Rao, S.; Patil, S.G. Particle Swarm Optimization based support vector machine for damage level prediction of non-reshaped berm breakwater. *Appl. Soft Comput. J.* **2015**, *27*, 313–321. [[CrossRef](#)]
22. Gedik, N. Least Squares Support Vector Mechanics to Predict the Stability Number of Rubble-Mound Breakwaters. *Water* **2018**, *10*, 12. [[CrossRef](#)]
23. Koc, M.L.; Balas, C.E.; Koc, D.I. Stability assessment of rubble-mound breakwaters using genetic programming. *Ocean Eng.* **2016**, *111*, 8–12. [[CrossRef](#)]
24. Mase, H.; Sakamoto, M.; Sakai, T. Neural Network for Stability Analysis of Rubble-Mound Breakwaters. *J. Waterw. Port Coast. Ocean Eng. ASCE* **1995**, *121*, 294–299. [[CrossRef](#)]
25. Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52. [[CrossRef](#)]
26. 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, Budapest, Hungary, 25–29 July 2004; Volume 982, pp. 985–990.
27. Rong, H.J.; Ong, Y.S.; Tan, A.H.; Zhu, Z.; Neucom, J. A fast pruned-extreme learning machine for classification problem. *Neurocomputing* **2008**, *72*, 359–366. [[CrossRef](#)]
28. Huang, G.B.; Ding, X.; Zhou, H. Optimization method based extreme learning machine for classification. *Neurocomputing* **2010**, *74*, 155–163. [[CrossRef](#)]
29. Pal, M.; Maxwell, A.E.; Warner, T.A. Kernel-based extreme learning machine for remote-sensing image classification. *Remote Sens. Lett.* **2013**, *4*, 853–862. [[CrossRef](#)]
30. Wei, L.; Chen, C.; Su, H.; Qian, D. Local Binary Patterns and Extreme Learning Machine for Hyperspectral Imagery Classification. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 3681–3693.
31. Cheng, L.; Zeng, Z.; Wei, Y.; Tang, H. Ensemble of extreme learning machine for landslide displacement prediction based on time series analysis. *Neural Comput. Appl.* **2014**, *24*, 99–107.
32. Abdullah, S.S.; Malek, M.A.; Abdullah, N.S.; Kisi, O.; Yap, K.S. Extreme Learning Machines: A new approach for prediction of reference evapotranspiration. *J. Hydrol.* **2015**, *527*, 184–195. [[CrossRef](#)]
33. Deo, R.C.; Şahin, M. Application of the extreme learning machine algorithm for the prediction of monthly Effective Drought Index in eastern Australia. *Atmos. Res.* **2015**, *153*, 512–525. [[CrossRef](#)]
34. Čojbašić, Ž.; Petković, D.; Shamsirband, S.; Tong, C.W.; Ch, S.; Janković, P.; Dučić, N.; Baralić, J. Surface roughness prediction by extreme learning machine constructed with abrasive water jet. *Precis. Eng.* **2016**, *43*, 86–92. [[CrossRef](#)]
35. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [[CrossRef](#)]
36. Huang, G.; Huang, G.B.; Song, S.; You, K. Trends in extreme learning machines: A review. *Neural Netw.* **2015**, *61*, 32–48. [[CrossRef](#)]

37. Alexandre, E.; Cuadra, L.; Nietoborge, J.C.; Candilgarcía, G.; Del Pino, M.; Salcedosanz, S. A hybrid genetic algorithm-extreme learning machine approach for accurate significant wave height reconstruction. *Ocean Model.* **2015**, *92*, 115–123. [[CrossRef](#)]
38. Yin, J.; Wang, N. An online sequential extreme learning machine for tidal prediction based on improved Gath-Geva fuzzy segmentation. *Neurocomputing* **2015**, *174*, 243–252.
39. Mulia, I.E.; Asano, T.; Nagayama, A. Real-time forecasting of near-field tsunami waveforms at coastal areas using a regularized extreme learning machine. *Coast. Eng.* **2016**, *109*, 1–8. [[CrossRef](#)]
40. Imani, M.; Kao, H.C.; Lan, W.H.; Kuo, C.Y. Daily sea level prediction at Chiayi coast, Taiwan using extreme learning machine and relevance vector machine. *Glob. Planet. Chang.* **2017**, *161*, S0921818117303715. [[CrossRef](#)]
41. Basheer, I.A.; Hajmeer, M. Artificial neural networks: Fundamentals, computing, design, and application. *J. Microbiol. Methods* **2000**, *43*, 3–31. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).