Máquina de aprendizado extremo com características selecionadas por um algoritmo evolutivo para a previsão de propriedades mecânicas de concretos de agregados leves

An extreme learning machine with features selected by an evolutionary algorithm for estimating mechanical properties of lightweight aggregate concretes

Abstract: In this paper, a Particle Swarm Optimization algorithm is used to adjust the parameters of an Extreme Learning Machine and select features in order to predict mechanical properties of lightweight aggregate concretes. Unlike the approaches found in the literature, the proposed procedure set the model parameters and select the most beneficial subset of features while simultaneously estimates two important outcomes: the compressive strength and elasticity modulus. These properties can be modeled as a function of up to four features: water/cement fraction, lightweight aggregate volume, cement quantity and lightweight aggregate density. The Particle Swarm Optimization algorithm performs the parameter and feature selection and automatically tunes the number of neurons in the hidden layer and the activation function. The results are compared with a model selection based on exhaustive search on the parameter space. The proposed approach arises as an alternative tool to select the most relevant features and to estimate the mechanical properties of lightweight aggregate concretes.

Keywords: Extreme Learning Machines, Particle Swarm Optimization, Lightweight Aggregate Concretes.

Data de submissão: 01/03/2018, Data de aceite: XX/XX/2018

1 Introduction

1 Reinforced concrete has been the most used structural material worldwide. As a result of rapid economic growth and urbanization, cement demand and cement production have grown rapidly over the last decades [1–3]. Moreover, when compared to other construction and building materials, reinforced concrete adapts itself to any kind of shape, allowing independence to the architectural conception, obtaining monolithic and hyperstatic structures, presenting excellent durability and low maintenance and preservation costs. On the other hand, a disadvantage is its own self-weight. In that context, the use of lightweight aggregate concretes emerges as an alternative solution. The lightweight aggregate concrete has refractory properties of thermal insulation and its specific weight is, approximately, two-thirds of the weight of the normal aggregate concrete. At the United States [4], for instance, the structural lightweight concrete is defined as the material with compressive strength over 17 MPa at 28 days and specific mass not over 1850 kg/m^3.

2 The specific weight of normal concretes varies between 2200 Kg/m^3 and 2600 Kg/m^3, and the structural lightweight concrete's between 1350 Kg/m^3 and 1850 Kg/m^3. Though it has disadvantages such as the greater cost, more care in placing, greater porosity and more drying shrinkage, the demand for lightweight aggregate concretes have been increasing [5]. The use of lightweight aggregates reduces

the dead load of a concrete structure, which then makes savings in reinforcement, footing and other load bearing elements. As an example, the structure's final cost can be lower due to the more economic scaling of the foundations.

Structures that use lightweight aggregate concrete as raw material need to be designed with high accuracy nowadays in order to avoid wastage of material, whereas. At the same time, there is a need for increasing its resistance to external stresses. Calculating the mechanical properties, such as the modulus
1 of elasticity and the compressive strength, becomes an important task, but also complicated for engineers, given the non-linearity of the relationship between the components that constitute the concrete and their properties. In this scenario, predicting the concrete properties with low cost and high reliability is a common goal of researchers in this area.

The relationship between concrete components and their mechanical properties is highly nonlinear, and the establishment of a comprehensive mathematical model is usually problematic. Standards propose expressions for the evaluation or lightweight aggregate concrete's modulus of elasticity (Young modulus, E) function of the characteristic compressive strength and density [4, 6] Most of the mathematical models used to study the behavior of concrete mixes consists on mathematical rules and expressions that capture
2 the relationship between components of concrete mixes. It is known that as the cement has a specific weight higher than the lightweight aggregate, for a given aggregate, the compressive strength increases with the increase of its specific mass [7]. Several studies have shown that the compressive strength is determined, regardless of the type of concrete, not only for the water to cement ratio but also for other components [8–14]. Considering the research scenario of the lightweight aggregate concretes, some studies have explored different predictive techniques. Artificial neural networks were used to predict the compressive strength of lightweight concretes in [15] and [16]. An artificial neural network approach was used to predict the compressive strength of lightweight and semi-lightweight concretes with pumice aggregate subjected to high temperatures in [17]. Multiple linear regression, clustering, decision trees and neural networks were evaluated to predict the modulus of elasticity in [18]. The development of the methodology proposed in this work, therefore, complements the studies currently found in the literature and fills this gap.

Feature selection has been a significant task for data mining and pattern recognition. The purpose of feature selection is to select the optimal feature subset that contains the most valuable information for making better decisions. In addition, it may work on a better comprehension of the domain, by maintain-
3 ing only the features with a good ability, according to some importance criterion [19] In feature selection, a determination of whether each feature is useful must be made, and the task of finding an optimal subset of features is inherently combinatory. Therefore, feature selection may become an optimization problem that requires an optimal approach to examine all possible subsets [20].

The purpose of this paper is to implement and evaluate the performance of Extreme Learning Machines combined with a Particle Swarm Optimization algorithm to predict two important mechanical properties of the lightweight aggregate concretes, the compressive strength and the modulus of elasticity (also called Young modulus). Experimental results show that Extreme Learning Machines obtained superior performance in terms of generalization and learning speed, in comparison to other algorithms in several real applications [21–23]. This model can provides researchers information regarding the mechanical behavior lightweight concrete without the need of performing destructive tests such as the usual compressive tests, reducing the number of laboratory tests and reworking.

2 Materials and Methods

2.1 Experimental Dataset

1 The experimental study [24] comprises lightweight aggregate concrete cylindrical specimens (16 × 32 cm) with two types of lightweight aggregate (shale and expanded clay), five levels of lightweight aggregate volume (0%, 12.5%, 25%, 37.5% and 45%) and three types of mortars (a, b, and c) as shown in Table 1. The elasticity modulus (E) and the compressive force (f_c) were measured for each sample (averaged over three measures). A total of 75 samples were collected and used to assess the performance of the approach proposed in this paper.

Table 1. Mix proportions of the lightweight aggregate concretes.

Mortar	Cement (kg/m^3)	Water/cement fraction	Aggregate density (kg/m^3)	Aggregate Volume (%)
Feature	x_2	x_0	x_3	x_1
	336.24		1055.46	0.0
	294.21		923.53	12.5
a	252.18	0.446	791.59	25.0
	210.15		659.66	37.5
	184.93		580.50	45.0
	283.98		1135.94	0.0
	248.49		993.95	12.5
b	212.99	0.350	851.95	25.0
	177.49		709.96	37.5
	156.19		624.77	45.0
	263.64		1074.38	0.0
	230.68		940.08	12.5
c	197.73	0.290	805.79	25.0
	164.77		671.49	37.5
	145.00		590.91	45.0

2.2 Extreme Learning Machines

2 The Extreme Learning Machine Regressor (ELM) is a feedforward single hidden layer artificial neural network with three levels of randomness [25], with some particular general characteristics as described as follow: (1) fully connected, hidden node parameters are randomly generated, (2) the connection can be randomly generated, not all input nodes are connected to a particular hidden node, and (3) a hidden node itself can be a subnetwork formed by several nodes resulting in learning local features.

The output function of ELM is given by

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{L} \beta_i G(\mathbf{w}_i, b_i, \mathbf{x}) = \sum_{i=1}^{L} \beta_i G\left(\sum_{k=1}^{D} w_{kj} x_k + b_i\right)$$

3 where $\hat{y}$ is the ELM prediction associated to the input vector $\mathbf{x}$, $\mathbf{w}_i$ is the weight vector of the i-th hidden layer, b_i are the biases of the neurons in the hidden layer, β_i are output weights, D is the number of input features, $G(\cdot)$ is the nonlinear activation function and L is the the number of neurons in the hidden layer. 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 presented in Table 2.

Table 2. Activation functions used in ELM. Independent of the training data, the hidden node parameters $(\mathbf{w}, b)$ are randomly generated using a normal distribution $N(0, 1)$ instead of being explicitly trained.

#	Name	Activation Function G
0	Identity	$G(\mathbf{w}, b, \mathbf{x}) = \mathbf{x}$
1	ReLU	$G(\mathbf{w}, b, \mathbf{x}) = \max(0, x_i; i = 1, \cdots, D)$
2	Multiquadric	$G(\mathbf{w}, b, \mathbf{x}) = \sqrt{\|\|\mathbf{w} - \mathbf{x}\|\|^2 + b^2}$
3	Sigmoid	$G(\mathbf{w}, b, \mathbf{x}) = \dfrac{1}{1 + exp(-\mathbf{w} \cdot \mathbf{x} + b)}$
4	Gaussian	$G(\mathbf{w}, b, \mathbf{x}) = exp(-(\mathbf{w} \cdot \mathbf{x} + b)^2)$
5	Inverse Multiquadric	$G(\mathbf{w}, b, \mathbf{x}) = \dfrac{1}{(\|\|\mathbf{w} - \mathbf{x}\|\|^2 + b^2)^{1/2}}$
6	Hyperbolic Tangent	$G(\mathbf{w}, b, \mathbf{x}) = \dfrac{1 - exp(\mathbf{w} \cdot \mathbf{x} + b)}{1 + exp(\mathbf{w} \cdot \mathbf{x} + b)}$

The output weight vector $[\beta_1, ..., \beta_L]$ can be determined by minimizing the approximation error [21]

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^L} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{y}\|$$

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

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

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

Figure 1 shows an example of a 4-8-2 ELM with four inputs, one hidden layer (8 neurons in each)
and two outputs. The input are water/cement fraction (x_0), lightweight aggregate volume (x_1), cement
1 quantity (x_2) and the lightweight aggregate density (x_3). The outputs are the elasticity modulus (E) and
the compressive strength (f_c).

2.3 Evolutionary settings of parameters

Setting the parameters of an estimator is usually a difficult task. Often, these parameters are
defined empirically, by testing different settings by hand. In this paper, we employ an evolutionary
2 algorithm to find the best set of parameters and the beneficial subset of features. The approach uses a
simple Particle Swarm Optimization where each particle is a representation of Extreme Learning Machine
parameters and a set of active features.

Particle swarm optimization (PSO) [26] is a stochastic population-based search method inspired
by the social behavior of animals such as birds and fish. In PSO, each individual, called a particle, flies
through the problem space and adjusts its position according to its own experience and the experience
3 of its neighbors. A particle can fly either fast and far from the best positions to explore unknown areas

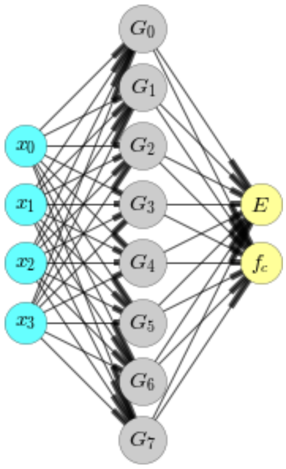

Figure 1. Connectivities for a 4-8-2 Extreme Learning Machine: four inputs, one hidden layer (8 neurons in each) and two outputs.

(global search), or very slowly and close to a particular position (fine tune) to find better results. Each
particle has a virtual position that represents a possible solution to some minimization problem. PSO is
1 quite simple to implement and has few control parameters. Equations (1) and (2) are the two fundamental
update rules of standard PSO.

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 (p_i(t) - x_i(t)) + c_2 r_2 (p_g(t) - x_i(t)) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

where v_i and x_i are velocity and position vectors of the particle i, respectively, p_i is the best local position
found by the particle i, and p_g is the best global position found in the whole population. The two param-
2 eters c_1 and c_2 are positive constants, called learning factors; c_1 presents how much a particle is attracted
to its best position, and c_2 is the same for the global position. Values of these two parameters may vary
depending on the nature of the problem but they are usually considered to be equal to 2.0; ω is the inertia
weight; r_1 and r_2 are uniform random variables providing the stochastic aspect of the algorithm.

In this paper, each particle encodes an ELM candidate solution. A particle's position represents
the number of neurons in the hidden layers, the activation function, and the set of selected features as
shown in Table 3. For ELM, the activation function G the number of hidden neurons HL, are necessary;
4 variables are required to determine whether each feature is selected or not. Therefore, 6 decision
3 variables must be identified. Considering the PSO approach, the goal is to find the decision variables,
corresponding to the ELM parameters and a subset of features, so that the network generates computed
outputs that match the outputs of the training data.

2.4 Cross-validation

Cross-validation is a sampling statistical technique to evaluate the ability of generalization of a
model from a dataset. Among the cross-validation techniques, k-fold [27] is one of the most used. k-
4 fold uses a part of the data available to fit the model, and another different part to test it. The dataset is
randomly divided into $k > 1$ subsets; from the k subsets, $k-1$ are used for training and the remaining

Table 3. Encoding of ELM candidate solutions. The column DV indicates the Decision Variable in the PSO encoding.

DV	Description	Range
0	Number of neurons in the hidden layer (see Fig. 1)	1–100
1	Coding representing the activation function according to Table 2	0: Identity, 1: ReLu; 2: Multiquadric, 3: Sigmoid, 4:Gaussian, 5: Inverse Multiquadric, 6: Hyperbolic Tangent
2	Decision variable corresponding to feature x_0	[0,1]; If a variable value is $\geq$ 0.5, then its corresponding feature is not chosen. Conversely, if a variable value is $>$0.5, then its corresponding feature is chosen.
3	Decision variable corresponding to feature x_1	[0,1]
4	Decision variable corresponding to feature x_2	[0,1]
5	Decision variable corresponding to feature x_3	[0,1]

set is used for testing. This process is repeated k times, using a different test set in each iteration. Figure 2 shows an example of 5-fold cross validation scheme.

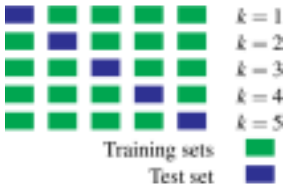

Figure 2. k-fold cross-validation method diagram ($k = 5$).

2.5 Performance Metrics

In this paper we have used the following metrics: R^2 score, Root Mean Squared Error (RMSE) and the Mean Absolute Percentage Error (MAPE). These criteria can be written as

$$\mathrm{R}^2 = 1 - \frac{\sum_{i=0}^{N-1}(y_i - \hat{y}_i)^2}{\sum_{i=0}^{N-1}(y_i - \bar{y})^2} \tag{3}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1}(y_i - \hat{y}_i)^2} \tag{4}$$

$$\mathrm{MAPE} = 100 \times \frac{1}{N}\sum_{i=0}^{N-1}\frac{|y_i - \hat{y}_i|}{|y_i|} \tag{5}$$

where $\hat{y}_i$ is the estimated target output, y_i is the corresponding target output, N is the number of samples, p is the number of model parameters, and $\bar{y}$ is the mean of the vector $[y_1, ..., y_N]^T$.

3 Computational Experiments

In this section, we present the results obtained for ELM+PSO approach described in Section 2.
In order to obtain consistent and reliable results, we ran each computational experiment 100 times using
5-fold cross-validation with shuffled data generated by different random seeds. In the model selection
1 step, PSO was set with the following parameters: 20 individuals in the population evolving under 35
generations; parameters c_1, c_2 were set to 2, respectively; $\omega = 0.6$ for all generations; the objective
function (to be minimized) is given by

$$\sqrt{\frac{1}{N}\sum_{i=0}^{N-1}(E_i - \hat{E}_i)^2 + \frac{1}{N}\sum_{i=0}^{N-1}(f_{c,i} - \hat{f_{c,i}})^2} \tag{6}$$

where E_i and $\hat{E}_i$ are the the measured and predicted elasticity modulus for sample i, respectively; $f_{c,i}$
2 and $\hat{f_{c,i}}$ are the the measured and predicted compressive strength for sample i, respectively; and N is the
number of samples. The range of parameters of ELM, number of hidden neurons and activation function
and the subset of features are shown in Table 3.

The computational experiments described here were conducted based in scikit-learn framework
[28] and implementations adapted from [29]and [30]. All codes and data are made available by the
3 authors upon request. Computer specifications used to execute ELM+PSO are given as follows: CPU
Intel i7 3.40GHz, RAM of 16 GB and operating system Linux Ubuntu 16.04.4 LTS.

Table 4. Elasticity modulus: comparison of ELM results considering MAPE, MSE, R^2 metrics.

Estimator	MAPE	MSE	R^2
RBF-ELM	2.369 ($\pm$0.206)	5.5e+05 ($\pm$1.0e+05)	0.983 ($\pm$0.003)
ELM+PSO	2.236 ($\pm$0.214)	5.2e+05 ($\pm$2.1e+05)	0.984 ($\pm$0.007)

Table 5. Compressive strength: comparison of ELM results considering MAPE, MSE, R^2 metrics.

Estimator	MAPE	MSE	R^2
RBF-ELM	6.883 ($\pm$0.764)	16.370 ($\pm$5.261)	0.943 ($\pm$0.018)
ELM+PSO	5.740 ($\pm$0.494)	11.972 ($\pm$4.552)	0.958 ($\pm$0.016)

Tables 4 and 5 provide a comparison of ELM+PSO results with a Radial Basis Function Extreme
Learning Machine (RBF-ELM) from [31]. The model selection in [31] comprises a exhaustive search
4 with cross-validation in a set of user defined parameters. It can be seen that ELM+PSO outperformed
RBF-ELM. We highlight the ELM+PSO estimates both E and f_c simultaneously. This feature may turn
the training procedure harder when compared with single models built for each output.

Figure 3 shows the distribution of activation functions (G) and number of hidden layer neurons
(HL). HL ranges from 15 to 60, whereas 20 is the most frequent values, chosen 18 out 100 runs; we
also observe that HL values between 16 and 23 appear 88 out 100 times. Analyzing the the distribution
5 of activation function parameter, the most frequent were Multiquadric (29 times), Sigmoid (28 times)
and Hyperbolic Tangent (24 times). Combined, Sigmoid, Hyperbolic Tangent and Multiquadric were
selected as activation functions in 81% of all independent runs. Summarizing, the model selection guided
by the PSO algorithm suggests an ELM network with 16-23 hidden neurons and implementing Sigmoid,
Hyperbolic Tangent or Multiquadric activation functions.

It should be noticed the former analysis does not consider the features selected along the PSO
6 training procedure. Figure 4 shows all subsets of features selected in the 100 independent runs: the

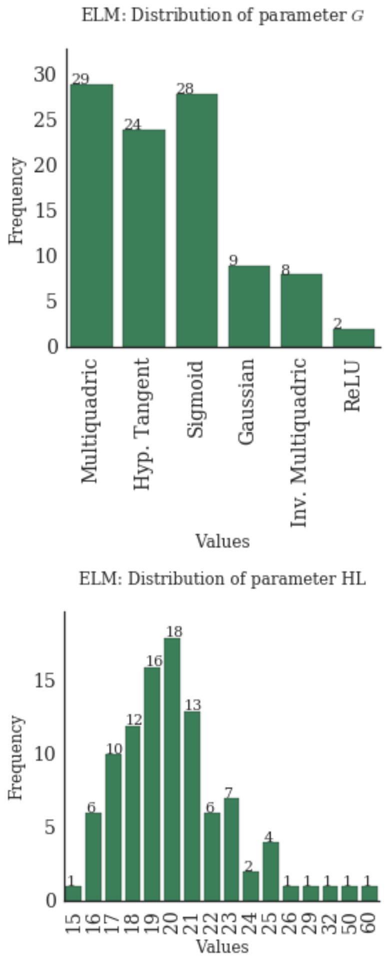

Figure 3. Distribution of parameters G and HL along 100 runs.

Table 6. Compressive strength: comparison of ELM results considering MAPE, MSE, R^2 metrics.

Output	Active Features	MAPE	R^2	RMSE
E	$x_0\|x_1\|x_2\|x_3$	2.227 (±0.265)	0.984 (±0.009)	707.674 (±144.930)
	$x_0\|x_1\|x_3$	2.236 (±0.232)	0.984 (±0.007)	714.114 (±122.434)
	$x_0\|x_2\|x_3$	2.241 (±0.139)	0.984 (±0.003)	706.443 (±57.272)
f_c	$x_0\|x_1\|x_2\|x_3$	5.841 (±0.419)	0.957 (±0.010)	3.507 (±0.389)
	$x_0\|x_1\|x_3$	5.696 (±0.581)	0.959 (±0.018)	3.370 (±0.562)
	$x_0\|x_2\|x_3$	5.724 (±0.412)	0.958 (±0.016)	3.422 (±0.537)

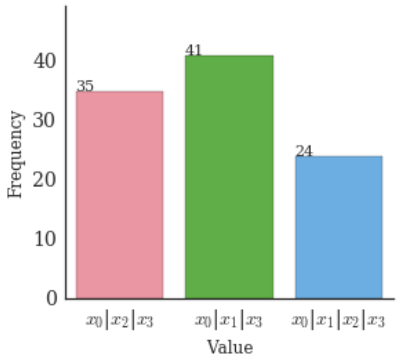

Figure 4. Distribution of the feature subsets along the 100 runs.

1 subset $x_0|x_1|x_3$ was selected 41 times, the subset $x_0|x_2|x_3$ 35 times while set involving all features $x_0|x_1|x_2|x_3$ was selected 24 times. In all runs the features x_0 and x_3 are included in the final solutions showing their relevance to predict the mechanical properties.

2 Table 6 presents the mean and standard deviations for MAPE, RMSE and R^2 according to the subset of selected features. The first column presents the output, the second column displays the set of active features, and the metrics are shown in remaining columns. We observe similar results for elasticity modulus (E) independent of the set of active features. For compressive strength (f_c), the subset $x_0|x_1|x_3$ produced slightly better results for MAPE and RMSE. One-way Analysis of variance (ANOVA) with a significance level of 5% were conducted to test whether the differences among the groups of selected features are statistically significant. In Table 7 the p-values ($p > 0.05$) provides sufficient evidence that the averages do not differ significantly for each metric (null hypothesis: the means are equal).

Table 7. One-way ANOVA with a significance level of 5% on the groups of selected features .

Output	Metric	F-value	p-value
E	MAPE	0.028	0.973
	R^2	0.114	0.892
	RMSE	0.050	0.951
f_c	MAPE	0.676	0.511
	R^2	0.217	0.805
	RMSE	0.519	0.597

3 Figure 5 shows the distribution of activation functions (G) and number of hidden layer neurons (HL) according to the subsets of features, detailing the results displayed in Figure 3. For the set of active features $x_0|x_2|x_3$, Multiquadric appears in the settings of the best estimator 14 out 35 times. Considering the set $x_0|x_1|x_3$, Sigmoid activation function was chosen as a parameter of the best estimator 14 out 41 times while the Multiquadric was selected 11 out 41 times. When all features ($x_0|x_1|x_2|x_3$) were included in the learning process, both Hyperbolic Tangent and Sigmoid produced good final solutions as can be seen in the number of times they were selected: Hyperbolic Tangent 9 out 24 runs and Sigmoid 7 out 24 runs. Independent of the groups of active features, Gaussian, Inverse Multiquadrics and ReLU activation functions do not produced good results since they were scarcely selected (less than 5 times out 100). As displayed in Figure 5, for the group of active features $x_0|x_2|x_3$ the number neurons in the hidden layer is most likely to be set in range [16-21] while for groups $x_0|x_1|x_3$ and $x_0|x_1|x_2|x_3$ set in the range [17-23]. We remark the ELM parameters and the set of active features were selected based on

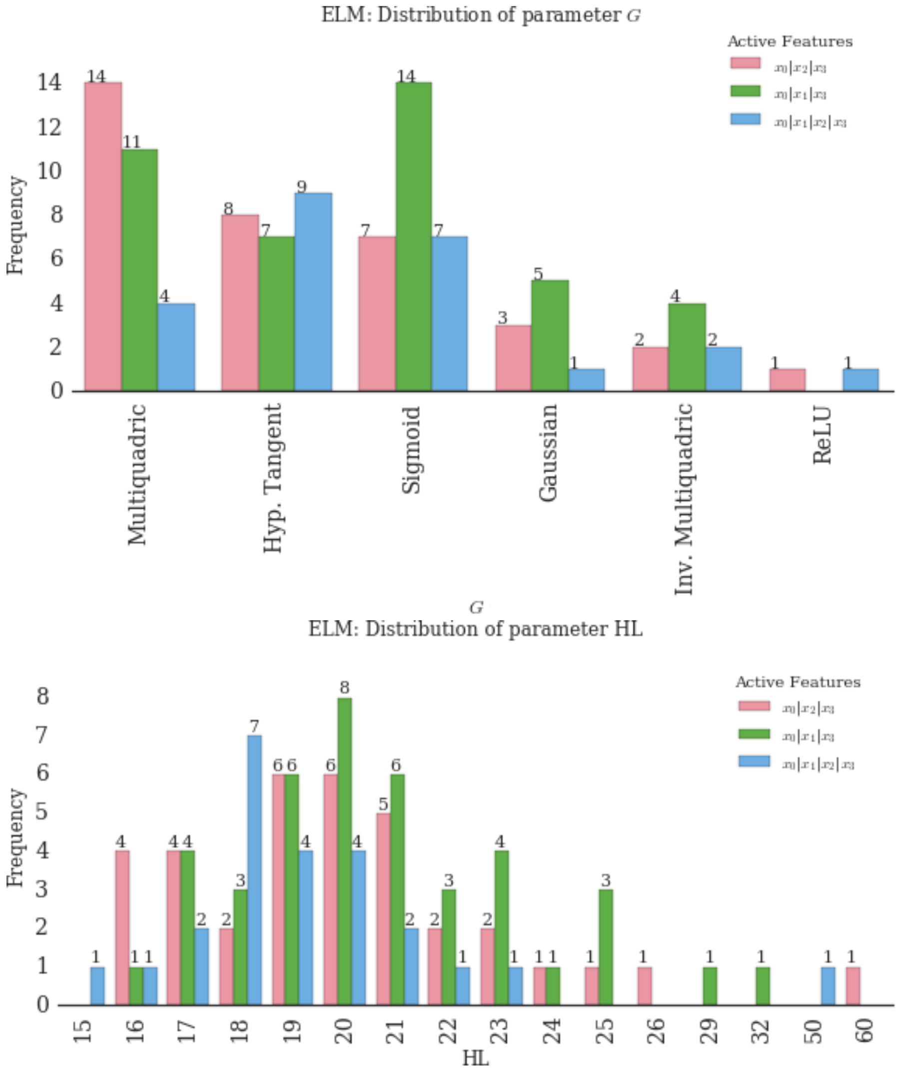

Figure 5. Distribution of parameters according to the feature subsets.

the minimization of Equation (6) used as PSO objective function.

It is also interesting to compare the computational cost of the evolutionary model selection with the exhaustive search procedure [28]. For a grid search procedure considering the range of 7 activations functions and 100 hidden neurons as suggested in this paper, a total of $7 \times 100 = 700$ model evaluations are required. Including the feature selection involving 4 features, 15 possible combinations were available: 1 combination involving 4 features, 4 involving 3 features, 6 involving 2 features and 4 with 1
1 feature. As a result, a total of $7 \times 100 \times 15 = 10500$ model evaluations are required to obtain the best model and the most relevant group of features. The PSO approach in this paper implements a population size of 20 individuals evolving along 35 generations, resulting in a total of $20 \times 35 = 700$ model evaluations. The grid search can become a very time-consuming step when the number of parameters increases or when we have a large set of user-defined options of each parameter. In such a case, the evolutionary approach for model selection arises as an interesting alternative to save computing time while producing optimal or near-optimal solutions for the model selection procedure.

4 Conclusions

In this paper, an Extreme Learning Machine is combined with a Particle Swarm Optimization algorithm to simultaneously predicting the elastic modulus and the compressive strength of lightweight aggregate concretes while selects the most relevant set of features. The proposed approach performs an intelligent search on the parameter space and arises as an interesting alternative to exhaustive search
2 allowing for computational savings in large sets of parameters. The results show good agreement with experimental results and competitive performance when compared with other machine learning approaches. Future works include evaluate other methods with a wide range of parameters and different estimation capabilities such as Multilayer Perceptron Neural Networks, Support Vector Machines and Gradient Boosting Regression.

*

References

1 GAO, T. et al. Evolution and projection of {CO2} emissions for china's cement industry from 1980 to 2020. *Renewable and Sustainable Energy Reviews*, v. 74, p. 522 – 537, 2017. ISSN 1364-0321.

2 SHEN, W. et al. Cement industry of china: Driving force, environment impact and sustainable development. *Renewable and Sustainable Energy Reviews*, v. 75, p. 618 – 628, 2017. ISSN 1364-0321.

3 IGLIńSKI, B.; BUCZKOWSKI, R. Development of cement industry in poland – history, current state, ecological aspects. a review. *Journal of Cleaner Production*, v. 141, p. 702 – 720, 2017. ISSN 0959-6526.

4 ACI. *Guide for structural lightweight aggregate concrete. ACI 213R-87.* [S.l.], 1999.

5 NILSON, A. H.; MARTINEZ, S. et al. Mechanical properties of high-strength lightweight concrete. In: *Journal Proceedings*. [S.l.: s.n.], 1986. v. 83, n. 4, p. 606–613.

6 CEN. *EN 1992-1-1 Eurocode 2: Design of concrete structures - Part 1-1: General ruels and rules for buildings*. Brussels, 2005.

7 BOGAS, J. A.; GOMES, A. Compressive behavior and failure modes of structural lightweight aggregate concrete – characterization and strength prediction. *Materials & Design*, v. 46, p. 832 – 841, 2013. ISSN 0261-3069. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0261306912007625>.

8 KASPERKIEWICZ, J.; RACZ, J.; DUBRAWSKI, A. Hpc strength prediction using artificial neural network. *Journal of Computing in Civil Engineering*, v. 9, n. 4, p. 279–284, 1995.

9 FAUST, T. Properties of different matrixes and lwas and their influences on the behavior of structural lwac. *Second int proceedings second international symposium on structural lightweight aggregate concrete*, p. 502 – 511, 2000.

10 NI, H.-G.; WANG, J.-Z. Prediction of compressive strength of concrete by neural networks. *Cement and Concrete Research*, v. 30, n. 8, p. 1245 – 1250, 2000. ISSN 0008-8846.

11 PAPADAKIS, V.; TSIMAS, S. Supplementary cementing materials in concrete: Part i: efficiency and design. *Cement and Concrete Research*, v. 32, n. 10, p. 1525 – 1532, 2002. ISSN 0008-8846.

12 CHANDRA, S.; BERNTSSON, L. *Lightweight aggregate concrete*. USA: Noyes Publications - William Andrew Publishing, 2003. (Science, technology and applications).

13 KOCKAL, N. U.; OZTURAN, T. Strength and elastic properties of structural lightweight concretes. *Materials and Design*, v. 32, n. 4, p. 2396 – 2403, 2011. ISSN 0261-3069.

14 DUAN, Z.; KOU, S.; POON, C. Using artificial neural networks for predicting the elastic modulus of recycled aggregate concrete. *Construction and Building Materials*, v. 44, n. 0, p. 524 – 532, 2013. ISSN 0950-0618.

15 ALTUN, F.; KIşI Özgür; AYDIN, K. Predicting the compressive strength of steel fiber added lightweight concrete using neural network. *Computational Materials Science*, v. 42, n. 2, p. 259 – 265, 2008. ISSN 0927-0256. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0927025607002017>.

16 ALSHIHRI, M. M.; AZMY, A. M.; EL-BISY, M. S. Neural networks for predicting compressive strength of structural light weight concrete. *Construction and Building Materials*, v. 23, n. 6, p. 2214 – 2219, 2009. ISSN 0950-0618.

17 BINGöL, A. F.; TORTUM, A.; GüL, R. Neural networks analysis of compressive strength of lightweight concrete after high temperatures. *Materials & Design (1980-2015)*, v. 52, p. 258 – 264, 2013. ISSN 0261-3069.

18 GARCíA Ángela et al. Prediction of mechanical strength of cork under compression using machine learning techniques. *Materials & Design*, v. 82, p. 304 – 311, 2015. ISSN 0264-1275.

19 DONG, H. et al. A novel hybrid genetic algorithm with granular information for feature selection and optimization. *Applied Soft Computing*, v. 65, p. 33 – 46, 2018. ISSN 1568-4946.

20 CHEN, S.-C.; LIN, S.-W.; CHOU, S.-Y. Enhancing the classification accuracy by scatter-search-based ensemble approach. *Applied Soft Computing*, v. 11, n. 1, p. 1021 – 1028, 2011. ISSN 1568-4946.

21 HUANG, G. et al. Trends in extreme learning machines: A review. *Neural Networks*, v. 61, n. Supplement C, p. 32 – 48, 2015. ISSN 0893-6080.

22 HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: a new learning scheme of feedforward neural networks. In: IEEE. *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*. [S.l.], 2004. v. 2, p. 985–990.

23 HUANG, G.-B. et al. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 42, n. 2, p. 513–529, 2012.

24 KE, Y. *Characterization of the mechanical behavior of lightweight aggregate concretes: Experiment and modelling*. Tese (Doutorado) — PhD. Thesis, Université de Cergy-Pontoise, 2008.

25 HUANG, G.-B. What are extreme learning machines? filling the gap between frank rosenblatt's dream and john von neumann's puzzle. *Cognitive Computation*, Springer, v. 7, n. 3, p. 263–278, 2015.

26 KENNEDY, J. Particle swarm optimization. In: *Encyclopedia of machine learning*. [S.l.]: Springer, 2011. p. 760–766.

27 HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning - Data Mining, Inference, and Prediction*. 2. ed. Verlag, New York: Springer, 2009. 745 p. ISBN 978-0-387-84857-0.

28 PEDREGOSA, F. et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, JMLR. org, v. 12, p. 2825–2830, 2011.

29 FRIEDMAN, J. H. Multivariate adaptive regression splines. *The annals of statistics*, JSTOR, p. 1–67, 1991.

30 PARTICLE swarm optimization (PSO) with constraint support. 2017. <https://github.com/tisimst/pyswarm/>. Accessed: 2017-12-20.

31 ANDRADE, J. J. et al. Aplicação de métodos de inteligência computacional para a previsão de propriedades mecânicas do concreto de agregado leve. In: *XXXVI Ibero-Latin American Congress on Computational Methods in Engineering*. [S.l.: s.n.], 2015.