Computers and Concrete, *Vol. 16, No. 4 (2015) 569-585*
DOI: http://dx.doi.org/10.12989/cac.2015.16.4.569

Nonlinear structural modeling using multivariate adaptive regression splines

Wengang Zhang and A.T.C. Goh*

School of Civil & Environmental Engineering, Nanyang Technological University, Block N1, Nanyang Avenue, 639798, Singapore

(Received November 26, 2012, Revised June 26, 2015, Accepted October 22, 2015)

Abstract. Various computational tools are available for modeling highly nonlinear structural engineering problems that lack a precise analytical theory or understanding of the phenomena involved. This paper adopts a fairly simple nonparametric adaptive regression algorithm known as multivariate adaptive regression splines (MARS) to model the nonlinear interactions between variables. The MARS method makes no specific assumptions about the underlying functional relationship between the input variables and the response. Details of MARS methodology and its associated procedures are introduced first, followed by a number of examples including three practical structural engineering problems. These examples indicate that accuracy of the MARS prediction approach. Additionally, MARS is able to assess the relative importance of the designed variables. As MARS explicitly defines the intervals for the input variables, the model enables engineers to have an insight and understanding of where significant changes in the data may occur. An example is also presented to demonstrate how the MARS developed model can be used to carry out structural reliability analysis.

Keywords: multivariate adaptive regression splines; structural analysis; nonlinearity; basis function; neural networks

1. Introduction

Many empirical and semiempirical methods expressed in the form of equations, tables or design charts, are commonly used in structural analysis and design. This is usually because of an inadequate understanding of the phenomena involved in the problem, as well as the complicated nonlinear multivariate nature of the problem. A typical example is the analysis of the behavior of deep Reinforced Concrete (RC) beams which has been the subject of numerous experimental and analytical studies. Deep beams have depths that are comparable to their span lengths. Because of the significant number of factors (parameters) that affect the behavior of deep beams and the complexity of behavior of these beams when subjected to shear failure, to date, the understanding of deep beam behavior is still limited.

For problems involving a large number of design (input) variables and nonlinear responses, particularly with statistically dependent input variables, an increasingly popular modeling

*Corresponding author, Professor, E-mail: ctcgoh@ntu.edu.sg

http://www.techno-press.org/?journal=cac&subpage=8 ISSN: 1598-8198 (Print), 1598-818X (Online)

technique is the use of neural networks. By far the most commonly used neural network model is known as the Back-propagation neural network (BPNN) algorithm (Rumelhart *et al*. 1986). A neural network has a parallel-distributed architecture with a number of interconnected nodes, commonly referred to as neurons. The neurons interact with each other via weighted connections. Each neuron is connected to all the neurons in the next layer. In the BPNN algorithm, neural network "learning" involves presenting a data pattern to the input layer, passing the signal through the intermediate layer where the input data is transformed via a nonlinear transfer function and determining the output (dependent variable). The processing of the inputs through the intermediate (hidden) neurons enables the network to represent and compute complicated associations between patterns. The main objective in "training" the neural network is to modify the connection weights to reduce the errors between the actual output values and the target output values through the minimization of the defined error function (e.g., sum squared error) using the gradient descent approach. Validation of the performance of the neural network is carried out by "testing" with a separate set of data that was never used in training the neural network, to assess the generalization capability of the trained neural network model to produce the correct input-output mapping even when the input is different from the examples used to train the network.

Neural networks have been successfully applied to a number of structural engineering problems including RC squat walls, RC deep beams and RC columns. Tsai (2010) proposed hybrid high order neural network model for predicting the strength of squat walls. A number of studies including Goh (1995), Sanad and Saka (2001), Jenkins (2006), Yang *et al*. (2008), Arafa *et al*. (2011) have demonstrated the feasibility of using neural networks to evaluate the ultimate shear strength of RC deep beams based on experimental results. These studies indicated that the predictions using neural networks were more accurate than those determined from conventional methods. Chuang *et al*. (1998) adopted a neural network model to predict the ultimate capacity of pin-ended RC columns under static loading. Oreta and Kawashima (2003) applied neural networks to predict the confined compressive strength and corresponding strain of circular concrete columns. Caglar (2009) developed a neural network model to determine the shear strength of circular RC columns. Alacali *et al*. (2011) established a neural network model to validate the empirical equations that are commonly used for prediction of the lateral confinement coefficient in RC columns.

One drawback of the BPNN is that it is computationally intensive. Typically training of the neural network to perform correctly requires thousands of iterations. A time-consuming trial-and-error approach is usually also necessary to find the optimal network architecture. Another limitation is the lack of model interpretability of the optimal network connection weights. Apart from the commonly used neural networks, other soft computing techniques applied in structural engineering problems include the genetic programming, the hybrid neuro-fuzzy approach, the hybrid coupling neural networks and simulated annealing method, etc. These related studies can be found in Gandomi *et al*. (2009), Gandomi *et al*. (2013), Alavi and Gandomi (2011).

This paper explores the use of an alternative procedure known as multivariate adaptive regression spline (MARS) (Friedman 1991) to model the nonlinear and multidimensional relationships. Previous applications of MARS approach in civil engineering include predicting doweled pavement performance (Attoh-Okine *et al*. 2009), modeling shaft resistance of piles in sand (Lashkari 2012), estimating deformation of asphalt mixtures (Mirzahosseinia *et al*. 2011), analyzing shaking table tests of reinforced soil wall (Zarnani *et al*. 2011), deriving undrained shear strength of clay (Samui and Karup 2011), and inferring ultimate capacity of driven piles in cohesionless soil (Samui 2011), uplift capacity of suction caisson in clay (Samui *et al*. 2011),

cavern serviceability limit state design (Zhang and Goh 2014), seismic liquefaction assessment (Zhang and Goh 2015) and lateral spreading induced by soil liquefaction (Goh and Zhang 2014). Zhang and Goh (2013) carried out extensive comparisons on the predictive performance between BPNN and MARS through six practical examples in geotechnical engineering.

The main advantages of MARS are its capacity to find the complex data mapping in high-dimensional data and produce simple, easy-to-interpret models, and its ability to estimate the contributions of the input variables. A number of examples are then presented to demonstrate the function approximating capacity of MARS and its efficiency in a noisy data environment, including three practical examples in structural engineering. An example is also presented to demonstrate how the MARS developed model can be used to carry out structural reliability analysis using Monte Carlo simulation.

2. MARS methodology

MARS is a nonlinear and nonparametric spline-based regression method that makes no specific assumption about the underlying functional relationship between the input variables and the output. The underlying idea behind MARS is to allow potentially different linear or nonlinear polynomial functions over different intervals. The end points of the intervals are called knots. A knot marks the end of one region of data and the beginning of another. The resulting piecewise curve (spline), gives greater flexibility to the model, allowing for bends, thresholds, and other departures from linear functions. An adaptive regression algorithm is used for selecting the knot locations. MARS models are constructed in a two-phase procedure. The first (forward) phase adds functions and finds potential knots to improve the performance, resulting in an overfit model. The second (backward) phase involves pruning the least effective terms. An open source code on MARS from Jekabsons (2011) is used in carrying out the analysis presented in this paper.

Let $\boldsymbol{y}$ be the target output and $\boldsymbol{X}$=(X_1, …, X_P) be a matrix of P input variables. Then it is assumed that the data are generated from an unknown 'true' model. In case of a continuous response this would be

$$y = f(X_1,\ldots X_p) + e = f(\mathbf{X}) + e \tag{1}$$

in which e is the distribution of the error. MARS approximates the function f by applying basis functions (BFs). BFs are splines (smooth polynomials), including piece-wise linear and piece-wise cubic functions. For simplicity, only the piece-wise linear function is expressed. Piece-wise linear functions are of the form max(0, $x-t$) with a knot occurring at value t. The equation max(.) means that only the positive part of (.) is used otherwise it is given a zero value. Formally

$$\max(0, x-t) = \begin{cases} x-t, if\ x \geq t \\ 0, otherwise \end{cases} \tag{2}$$

The MARS model, $f(\boldsymbol{X})$, is constructed as a linear combination of BFs and their interactions, and is expressed as

$$f(X) = \beta_0 + \sum_{m=1}^{M} \beta_m \lambda_m(X) \tag{3}$$

where each λ_m is a basis function. It can be a spline function, or the product of two or more spline

functions already contained in the model (higher orders can be used when the data warrants it; for simplicity, at most second order is assumed in this paper). The coefficients β are constants, estimated using the least-squares method.

The MARS modeling is a data-driven process. To fit the model in Eq. (3), first a forward selection procedure is performed on the training data. A model is constructed with only the intercept, β_0, and the basis pair that produces the largest decrease in the training error is added. Considering a current model with M basis functions, the next pair is added to the model in the form

$$\hat{\beta}_{M+1}\lambda_m(X)\max(0, X_j - t) + \hat{\beta}_{M+2}\lambda_m(X)\max(0, t - X_j) \tag{4}$$

with each β being estimated by the method of least squares. As a basis function is added to the model space, interactions between BFs that are already in the model are also considered. BFs are added until the model reaches some maximum specified number of terms leading to a purposely overfit model. To reduce the number of terms, a backward deletion sequence follows.

The aim of the backward deletion procedure is to find a close to optimal model by removing extraneous variables. The backward pass prunes the model by removing terms one by one, deleting the least effective term at each step until it finds the best sub-model. Model subsets are compared using the less computationally expensive method of Generalized Cross-Validation (GCV). The GCV equation is a goodness of fit test that penalize large numbers of BFs and serves to reduce the chance of overfitting. For the training data with N observations, GCV for a model is calculated as follows (Hastie *et al*. 2009)

$$GCV = \frac{\frac{1}{N}\sum_{i=1}^{N}[y_i - f(x_i)]^2}{[1 - \frac{M + d \times (M-1)/2}{N}]^2} \tag{5}$$

in which M is the number of BFs, d is the penalizing parameter and N is the number of data sets, and $f(x_i)$ denotes the predicted values of the MARS model. The numerator is the mean square error of the evaluated model in the training data, penalized by the denominator. The denominator accounts for the increasing variance in the case of increasing model complexity. Note that $(M-1)/2$ is the number of hinge function knots. The GCV penalizes not only the number of model's basis functions but also the number of knots. A default value of 3 is assigned to penalizing parameter d (Friedman 1991). At each deletion step a basis function is removed to minimize Eq. (5), until an adequately fitting model is found. MARS is an adaptive procedure because the selection of BFs and the variable knot locations are data-based and specific to the problem at hand.

After the optimal MARS model is determined, by grouping together all the BFs that involve one variable and another grouping of BFs that involve pairwise interactions (and even higher level interactions when applicable), this procedure called the analysis of variance (ANOVA) decomposition (Friedman 1991) can be used to assess the contributions from the input variables and the BFs.

3. Analyses using MARS

Some examples are presented to illustrate the application and accuracy of MARS. The cowboy

hat surface function has been widely used for validating the performance of regression and neural network models. The RC squat wall example tests the predictive capacities of MARS model in estimating the peak shear strength. The deep beam example is used to examine the capabilities of MARS for prediction of shear strength. The RC column example models the ultimate capacity under static loading. For the deep beam example, it is also demonstrated that the MARS developed model can be used to carry out structural reliability analysis.

MARS predictions are compared with the neural networks, including conventional BPNN and Evolutionary Bayesian Back-propagation (EBBP) proposed by Chua and Goh (2003). The EBBP is a modification of the Bayesian back propagation neural network proposed by Mackay (1991) and Neal (1992) which simplifies the network architecture selection by constraining the size of the network parameters through a regularizer that penalizes the more complicated weight functions in favor of simpler functions by adding a penalty term to the sum squared error. The main enhancement in the EBBP is the incorporation of the genetic algorithms search technique to determine the optimal weights.

3.1 Cowboy hat surface

Fig. 1 shows a cowboy hat surface function that has been widely used for validating the performance of regression and neural network models. Both x_1 and x_2 are limited to [-3, 3]. A set of data points consisting of 500 training data and 300 testing data were randomly generated using uniform distributions for x_1 and x_2, respectively. The values of z are then calculated from the Equation $z=\sin\left(\sqrt{x_1^2+x_2^2}\right)$. Chua (2001) found that the EBBP predicts well in terms of *MSE* especially for the testing phase.

To evaluate the accuracy of MARS, the same problem is considered. In the first (forward) phase, a maximum number of 70 BFs of linear spline function with second-order interaction were specified and subsequently 28 BFs were pruned from the final MARS model in the second

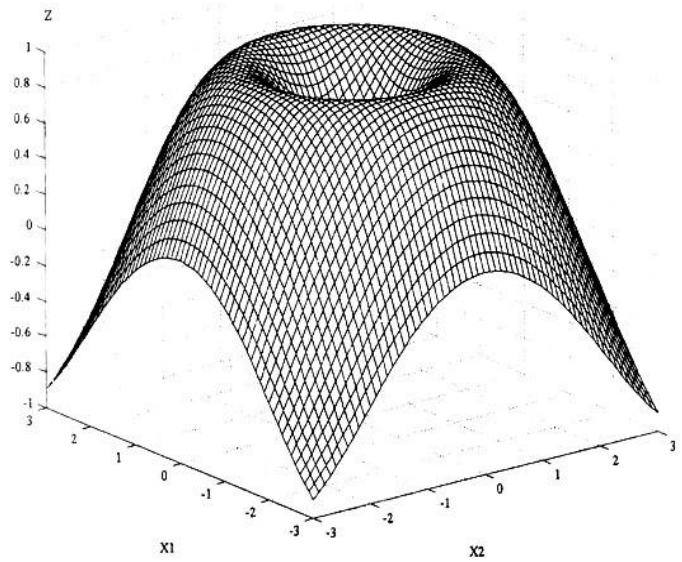
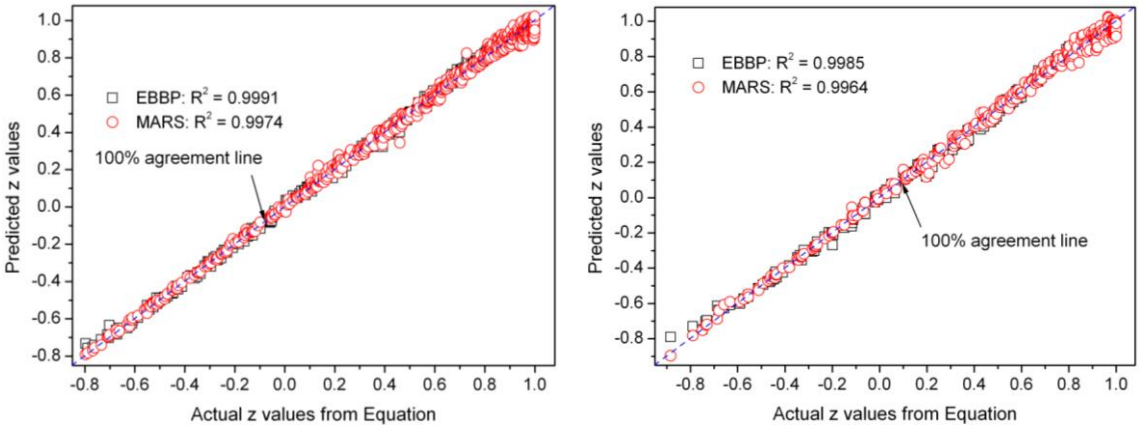


Fig. 1 Cowboy hat surface

Table 1 Comparison of results from EBBP and MARS for fitting cowboy hat

Methods	Training phase	Testing phase
EBBP		
MSE ($\times 10^{-3}$)	0.4	0.7
the coefficient of determination R^2	0.9991	0.9985
MARS		
MSE ($\times 10^{-3}$)	0.6	0.8
the coefficient of determination R^2	0.9974	0.9964



(a) training data (b) testing data

Fig. 2 Prediction of cowboy hat function by MARS and EBBP

(backward) phase. The execution time is 37.30s. The summary of the predictions is shown in Table 1. Fig. 2 shows the predictions given by MARS and EBBP. Generally, MARS performs as well as, if not better than the EBBP in terms of *MSE* especially for the testing phase. In addition, MARS is computationally efficient in terms of processing speed.

3.2 RC squat wall analysis

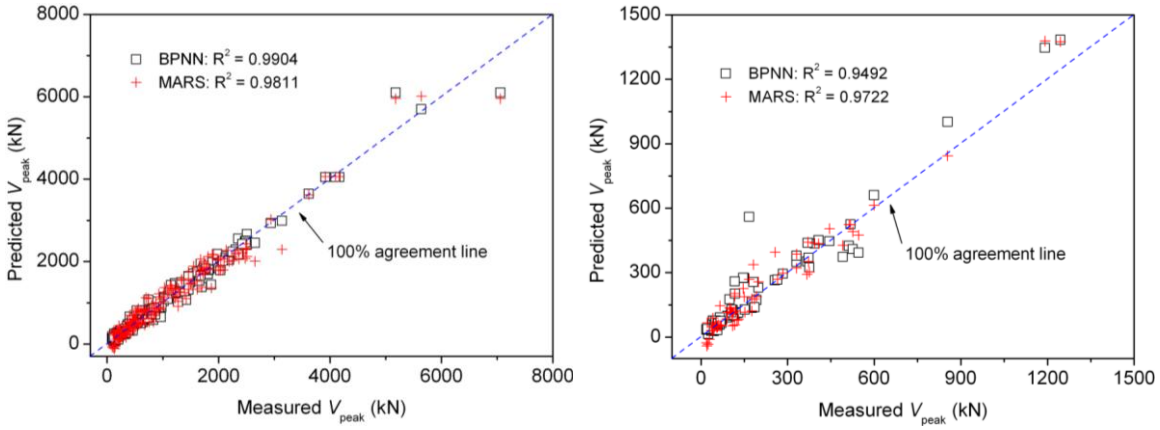
Short (squat) reinforced concrete walls are walls with a ratio of height to length of less than two and generally grouped by plan geometry, namely, rectangular, barbell, and flanged. Accurate modeling of the peak shear strength of squat walls is important because they would provide much or all of a structure's lateral strength and stiffness to resist seismic effects and wind loadings. Tsai (2011) developed a weighted genetic programming approach to study the squat wall strength and the results demonstrated that the proposed method provided accurate predictions and formula outputs. In this paper, the extensive experimental database compiled by Gulec (2009) was used to determine the peak shear strength of squat walls with barbell and flanged cross-sections.

The database adopted in this study consisted of 284 experimental cases. A total of nine input variables comprising the geometric and reinforcement parameters, material properties and loading types are assumed in this study. A summary of the input variables and outputs is listed in Table 2.

Of the 284 experimental test results, 213 samples were randomly selected as the training data

Table 2 Statistical parameters of input and output variables for RC squat walls

Variable	Parameters	Physical meaning	Ranges
1	t_w (m)	thickness of wall web	0.05-0.2
2	h_w (m)	height of wall	0.40-2.62
3	l_w (m)	length of wall	0.51-3.96
4	M/Vl_w	moment-to-shear ratio	0.06-1.9
5	ρ_v (%)	vertical web reinforcement ratio	0-2.8
6	ρ_{vall} (%)	ratio of total area of vertical reinforcement to wall area	0.44-5.91
7	ρ_h (%)	horizontal web reinforcement ratio	0-2.8
8	$f_c^{'}$ (kPa)	compressive strength of concrete	10005-104004
9	T	Loading type: 1 for cyclic; 2 for monotonic; 3 for dynamic; 4 for repeated; 5 for blast	1-5
Output V_{peak} (kN)		the peak shear strength of squat walls	85-7060



(a) training data (b) testing data

Fig. 3 Performance of MARS model for predicting the peak shear strength

and the remaining 71 data samples were used for testing. The data sets used for training and testing can be referred to Gulec (2009). Based on a trial-and-error approach, the derived optimal BPNN model consisted of five hidden neurons.

Using the same training samples, the MARS model consisted of 10 BFs of linear spline functions with second-order interaction. The execution time of 1.05s indicates that MARS model is computationally efficient in terms of processing speed. A plot of the BPNN and MARS predicted V_{peak} values versus the measured values for the training and testing patterns are shown in Fig. 3. Comparison between BPNN and MARS shows that the BPNN model is only slightly more accurate than the MARS model for the training patterns. For the testing results, the MARS model performs slightly better than the BPNN model. Therefore, both MARS and BPNN can serve as reliable tools for the prediction of the peak shear strength.

Table 3 displays the ANOVA decomposition of the developed MARS models. The first column in Table 3 lists the ANOVA function number. The second column gives an indication of the importance of the corresponding ANOVA function, by listing the GCV score for a model with all

Table 3 ANOVA decomposition of the developed MARS model for RC squat walls

Functions	*GCV*	*STD*	#basis	variable(s)
1	772099	319.6	1	t_w
2	689686	496.9	1	l_w
3	128393	241.9	1	M/Vl_w
4	265117	154.8	1	ρ_v
5	254180	269.5	1	$f_c^{'}$
6	113740	223.3	1	$t_w \rho_{vall}$
7	158896	240.2	2	$l_w f_c^{'}$
8	377393	502.1	1	$M/Vl_w f_c^{'}$
9	62587	83.5	1	$\rho_v \rho_{vall}$

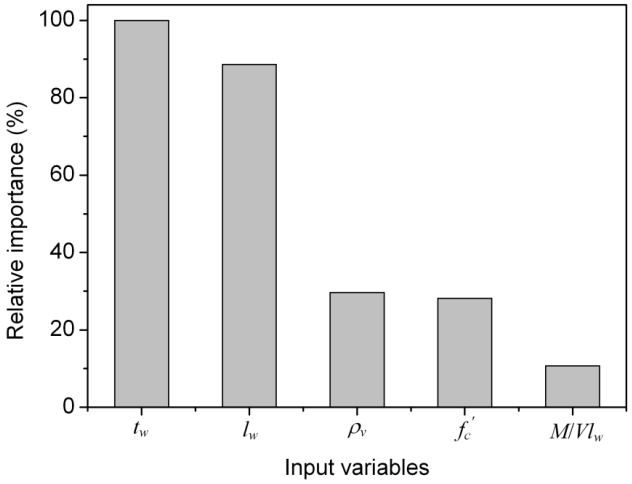


Fig. 4 Relative importance of the input variables selected in the MARS model

BFs corresponding to that particular ANOVA function removed. This GCV score can be used to evaluate whether the ANOVA function is making an important contribution to the model, or whether it just marginally improves the global GCV score. The third column provides the standard deviation of this function. The fourth column gives the number of BFs comprising the ANOVA function. The last column gives the particular input variables associated with the ANOVA function. Fig. 4 shows the plots of the relative importance of the input variables, which is evaluated by the increase in the GCV value caused by removing the considered variables from the developed MARS model. It can be observed that the thickness of the wall t_w is the most important parameter, followed by the wall length l_w and the vertical web reinforcement ratio ρ_v.

Table 4 lists the BFs of the MARS model and their corresponding equations. For the expression of BFs 7-10, F'_c is normalized between 0.1 and 0.9 through $F'_c=0.1+(f'_c-f'_{cmin})/(f'_{cmax}-f'_{cmin})\times 0.8$. The interpretable MARS model to predict the peak shear strength is given by

$$V_{peak}(kN) = 2552 - 10014\times BF1 - 866.2\times BF2 - 303.2\times BF3 - 2.23\times 10^5 \times BF4 + 6187\times BF5 + 122.5\times BF6 - 4563\times BF7 + 449.6\times BF8 + 3840\times BF9 + 2.1\times 10^5 \times BF10 \quad (6)$$

Table 4 Basis functions and their corresponding equations for RC squat walls

Basis function	Equation
BF1	$\max(0, 0.15 - t_w)$
BF2	$\max(0, 2.30 - l_w)$
BF3	$\max(0, 2 - \rho_v)$
BF4	$\max(0, t_w - 0.15) \times \max(0, 0.84 - \rho_{vall})$
BF5	$\max(0, 0.34 - M/Vl_w)$
BF6	$BF3 \times \max(0, \rho_{vall} - 1.04)$
BF7	$\max(0, 0.316 - F'_c)$
BF8	$BF2 \times \max(0, F'_c - 0.319)$
BF9	$BF2 \times \max(0, 0.319 - F'_c)$
BF10	$\max(0, F'_c - 0.316) \times \max(0, 0.55 - M/Vl_w)$

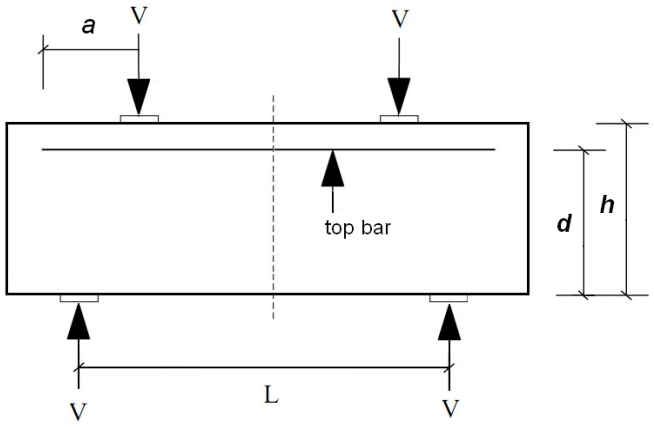


Fig. 5 Deep beam configuration

3.3 Deep beam analysis

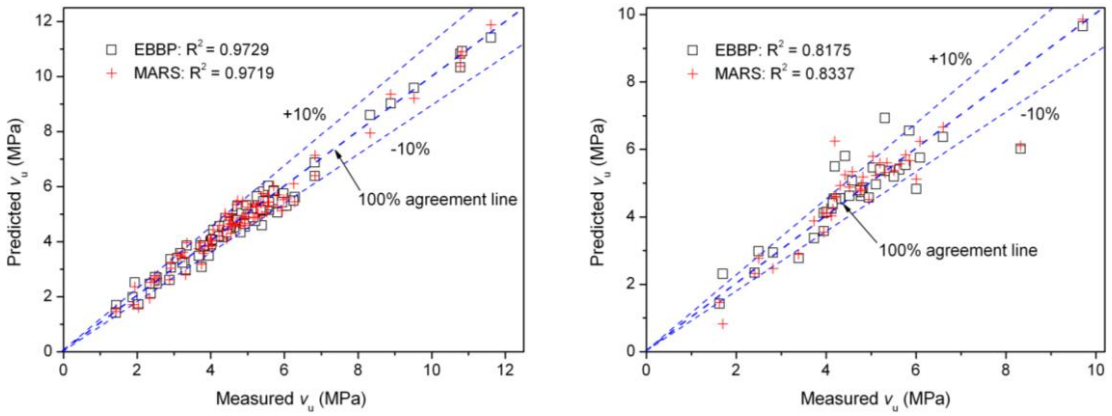
Deep beam design is of considerable importance in structural engineering. Deep beams have depths that are comparable to their span lengths. The behavior of deep RC beams has been the subject of numerous experimental and analytical studies. Due to a great number of factors influencing the behavior of deep beams and the complexity of behavior of these beams when subjected to shear failure, the understanding of deep beam behavior is limited. Several design methods have been proposed, each based on differing assumptions and concepts. It is beyond the scope of this paper to discuss these conventional design methods. The basic parameters of the deep beam are shown in Fig. 5.

In this example, the experimental database used in the EBBP analysis by Goh and Chua (2004) was reanalyzed using MARS. The database consisted of 90 observations for training and 38 observations for testing. The EBBP architecture consisted of six input neurons, six hidden neurons and one output neuron representing the ultimate shear strength v_u. The range of the six input parameters is summarized in Table 5.

The deep beam analysis using MARS adopted 16 BFs of linear spline functions with second-

Table 5 Statistical parameters of input variables for deep beams

Parameters	Physical meaning	Range
a (mm)	shear span	121.9-1292
d (mm)	effective depth	215.9-950
$f_c^{'}$ (MPa)	cylinder compressive strength of concrete	12.3-39.0
ρ_{ht} (%)	reinforcement ratio of the total horizontal steel	0.012-3.36
ρ_h (%)	reinforcement ratio of the horizontal tensile steel	0-2.45
ρ_v (%)	reinforcement ratio of the transverse steel	0-2.45



(a) training data (b) testing data

Fig. 6 Predicted versus measured values for deep beams

order interaction. The execution time for MARS was 1.31s. A plot of the EBBP predicted and MARS predicted v_u values versus the measured values for the training data patterns is shown in Fig. 6a. Most of training data fall within the ±10% error line. As shown in the plot of the testing data in Fig. 6b, the MARS predictions are as accurate as the EBBP. The ratios of the predicted strength to the measured strength of the 38 testing patterns for MARS are shown in Table 6 together with EBBP predictions. The results clearly demonstrate the accuracy of MARS except for three cases italicized in the table (test No. 18, 37 & 38). Both MARS and EBBP do not give good predictions for these three data points possibly because of actual measurement errors.

The ANOVA parameter relative importance assessment indicates that the two most important variables are $f_c^{'}$ (the compressive strength of concrete) and a (the shear span). For brevity, the ANOVA decomposition data has been omitted. Table 7 lists the BFs and their corresponding equations. It is observed from Table 7 that of the 16 basis functions, 10 BFs with interaction terms are integrated in this optimal model (BF6, BF7, BF8, BF9, BF10, BF12, BF13, BF14, BF15 and BF16), indicating that the model is not simply additive and that interactions play an important role. The MARS developed equation for predicting the ultimate shear strength of deep beams v_u is

$$\begin{aligned} v_u(MPa) &= 6.63 - 0.0093\times BF1 + 0.0714\times BF2 + 1.382\times BF3 - 0.1016\times BF4 \\ &+1.5744\times BF5 - 0.6919\times BF6 - 6.3877\times BF7 + 1.2\times 10^{-5}\times BF8 - 2.5629\times BF9 \\ &-0.0034\times BF10 + 1.8505\times BF11 - 0.0793\times BF12 - 1.0761\times BF13 - 6.6\times BF14 \\ &-0.0014\times BF15 - 0.0045\times BF16 \end{aligned} \tag{7}$$

Table 6 Predicted shear strength results for deep beam testing data

Testing No.	Measured strength / Predicted strength	
	EBBP	MARS
1	1.244	1.174
2	1.060	0.976
3	0.762	0.843
4	0.891	1.033
5	1.036	0.991
6	1.014	0.994
7	1.043	1.011
8	1.062	1.026
9	1.043	0.988
10	1.030	0.978
11	1.108	0.963
12	0.968	0.978
13	0.943	0.980
14	0.838	0.901
15	1.224	1.171
16	1.148	1.119
17	0.766	1.000
18	*1.384*	*1.364*
19	0.996	0.998
20	0.971	1.023
21	0.961	0.960
22	1.036	0.965
23	0.979	0.927
24	0.999	0.955
25	0.922	0.870
26	1.007	0.985
27	0.903	0.858
28	0.946	0.875
29	0.989	0.930
30	0.962	0.930
31	1.083	1.097
32	0.919	0.905
33	0.932	0.920
34	0.961	1.142
35	1.029	1.039
36	1.103	1.114
37	*0.763*	*0.671*
38	*0.737*	*2.059*
Average	0.994 (1.000, if No. 37 & 38 omitted)	1.019 (1.007)

Table 7 Basis functions and their corresponding equations for deep beams

Basis function	Equation
BF1	max(0, a – 216.54)
BF2	max(0, 216.54 – a)
BF3	max(0, $f_c^{'}$ – 30.13)
BF4	max(0, 30.13 – $f_c^{'}$)
BF5	max(0, ρ_{ht} – 0.94)
BF6	BF3 × max(0, ρ_v – 0.09)
BF7	BF3 × max(0, 0.09 – ρ_v)
BF8	BF1 × max(0, d – 550)
BF9	BF5 × max(0, 0.77 – ρ_v)
BF10	BF5 × max(0, 600 – a)
BF11	max(0, 0.56 – ρ_v)
BF12	BF11 × max(0, 234.7 – a)
BF13	max(0, 0.94 – ρ_{ht}) × max(0, ρ_v – 0.02)
BF14	BF11 × max(0, 0.94 – ρ_{ht})
BF15	BF3 × max(0, a – 588)
BF16	BF3 × max(0, 525 – d)

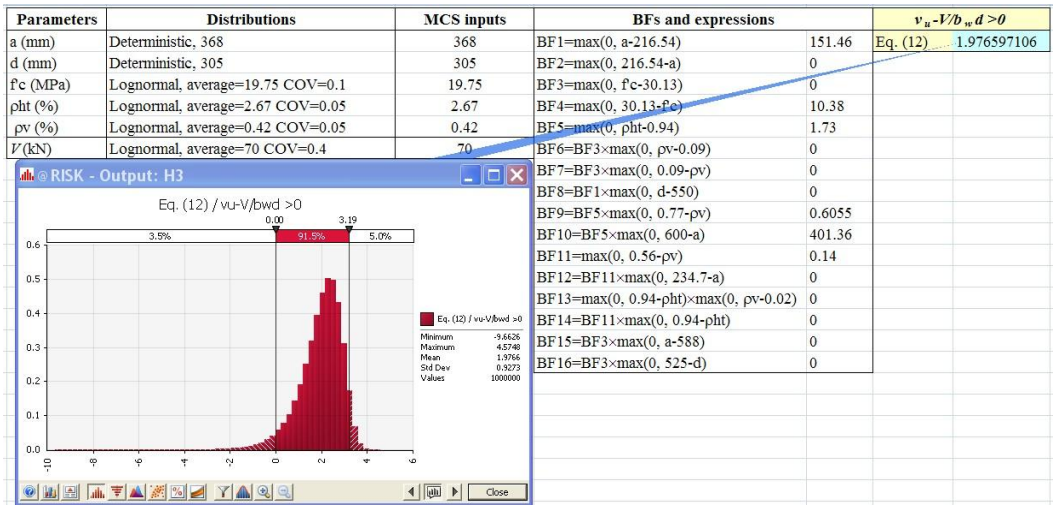


Fig. 7 Implementation of MARS model into MCS for reliability analyses

With the determination of the performance function Eq. (7), reliability assessment of the ultimate shear strength can be performed using Monte Carlo Simulation (MCS), as shown in Fig. 7. Failure occurs if the predicted ultimate shear strength v_u is smaller than the applied shear stress defined as $V/b_w d$, in which b_w is the breadth of the beam. The MCS starts with the characterization of the probability distributions (assumed as lognormals in this example) of the random variables (the applied load, the compressive strength of concrete and the reinforcement ratios), followed by the generation of predetermined sets of random samples. The statistical information of the input

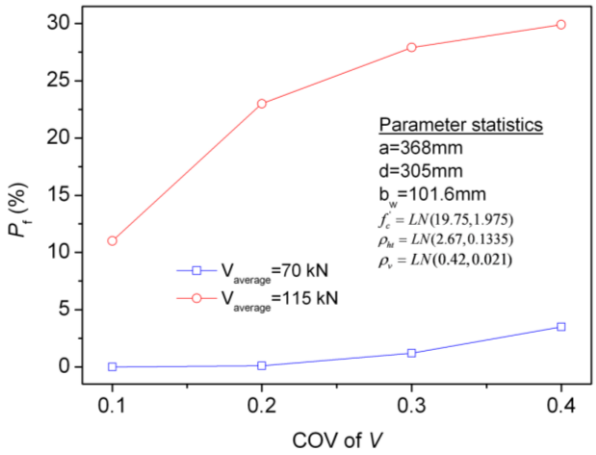


Fig. 8 Influence of COV of V on P_f

Table 8 Statistical parameters of input and output variables for RC columns

Variable No.	Parameters	Physical meaning	Ranges
1	b (mm)	width of the cross section	25-400
2	h (mm)	depth of the cross section	10-251
3	d/h	relative depth of tension steel reinforcement	0.7-0.94
4	100ρ	the reinforcement ratio	0.5-5.61
5	f_{cu} (MPa)	concrete cube strength	16.7-55
6	f_y (MPa)	steel yield strength	206-530
7	e/h	relative load eccentricity	0-3.2
8	L/h	relative overall length	8.8-60
Response N_u (kN)		column ultimate capacity	9.8-2040

variables is listed in Fig. 8.

For illustrative purposes, the effect of the coefficient of variation (COV) of the applied load V ranging from 0.1 to 0.4 is investigated. The P_f in Fig. 8 is the probability that the predicted ultimate shear strength v_u is smaller than the shear stress induced by the applied load V. The results indicate that both the COV and the average value of load V significantly influence the P_f.

3.4 Modeling behavior of RC columns

For the RC column analysis to determine the ultimate capacity of pin-ended RC columns under static loading using MARS, the results are compared with the neural network (BP8) analysis carried out by Chuang *et al*. (1998). The network structure of BP8 is three-layered with 12 hidden neurons in the hidden layer. The input layer consists of eight neurons representing eight parameters as shown in Table 8. The geometrical properties of the concrete column are illustrated in Fig. 9. The output layer consists of one neuron representing the ultimate capacity of the column N_u. Table 8 summarizes the range of values for all the parameters in the experimental database. A total of 45 of the 226 tests were selected as the testing data, and the remaining 181 tests were for model training.

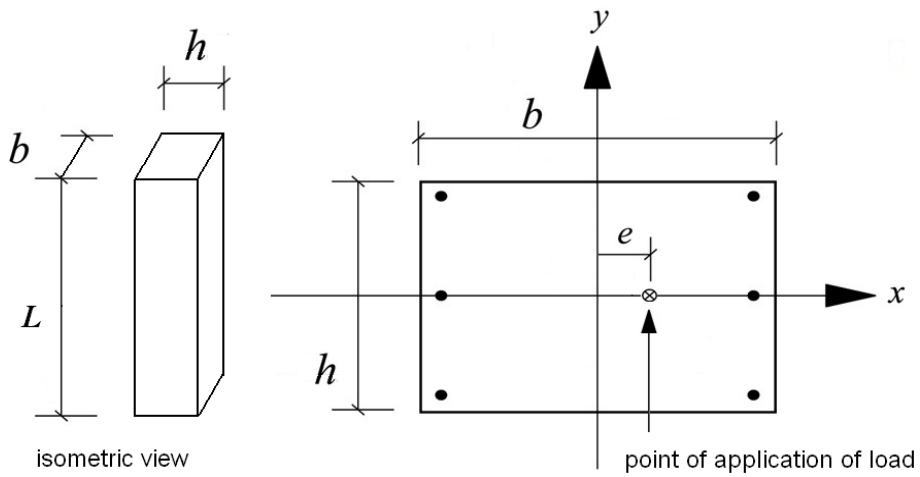
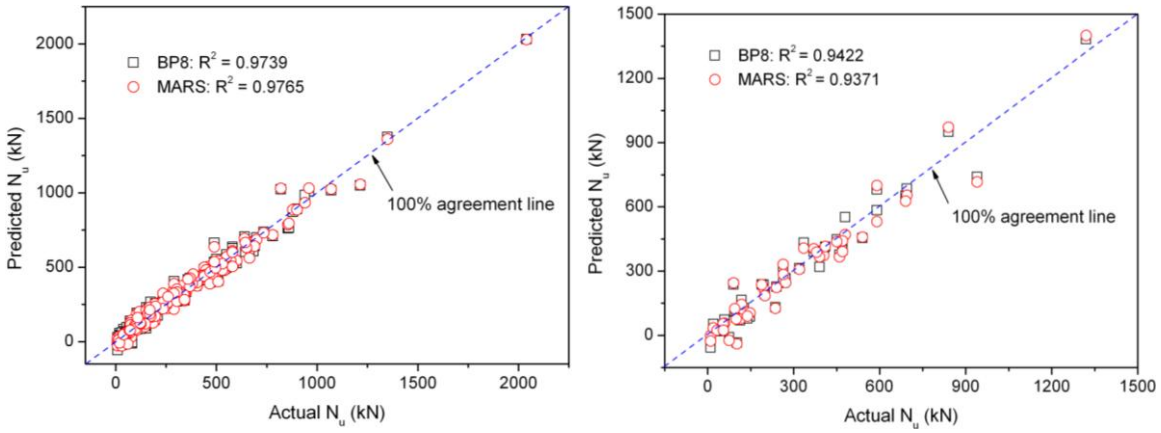


Fig. 9 Typical geometry of RC columns



(a) training data (b) testing data

Fig. 10 Predicted versus measured values for RC columns

The pin ended RC column analysis using MARS adopted 22 BFs of linear spline functions with second order interaction. The execution time of 3.73s shows that MARS is computationally very fast. A plot of BP8 and MARS predicted values versus the measured values for the training and testing data patterns is shown in Fig. 10. Comparison with the measured testing data in terms of R^2 shows that the ultimate capacity of reinforced concrete columns predicted by BP8 and MARS models are reasonably accurate.

The ANOVA parameter relative importance assessment indicates that the two most significant variables are *h* (depth of the cross section) and *b* (width of the cross section). For brevity, the ANOVA decomposition data has been omitted. Table 9 lists the BFs and their corresponding equations. It is noted from Table 9 that of the 22 basis functions, 19 BFs with interaction terms are integrated in this model (excluding BF1, BF8 and BF15), indicating that the model is not simply additive and that interactions play a significantly important role. The interpretable MARS model is given by

Table 9 Basis functions and their corresponding equations for RC columns

Basis function	Equation
BF1	max(0, 76 – *h*)
BF2	max(0, *h* – 76) × max(0, *e*/*h* – 0.25)
BF3	max(0, *h* – 76) × max(0, 40 – *L*/*h*)
BF4	BF1 × max(0, *L*/*h* – 21.9)
BF5	BF1 × max(0, 21.9 – *L*/*h*)
BF6	max(0, *h* – 76) × max(0, 182 – *b*)
BF7	BF1 × max(0, 0.25 – *e*/*h*)
BF8	max(0, 70 – *b*)
BF9	max(0, *b* – 70) × max(0, 3.25 – 100ρ)
BF10	max(0, *b* – 70) × max(0, *h* – 178)
BF11	max(0, 0.25 – *e*/*h*) × max(0, *h* – 152)
BF12	max(0, *h* – 76) × max(0, 0.87 – *d*/*h*)
BF13	max(0, *e*/*h* – 0.25) × max(0, 29.4 – *L*/*h*)
BF14	max(0, *b* – 70) × max(0, 0.25 – *e*/*h*)
BF15	max(0, 24.1 – f_{cu})
BF16	max(0, 0.25 – *e*/*h*) × max(0, *L*/*h* – 23.8)
BF17	max(0, f_{cu} – 24.1) × max(0, 12.6 – *L*/*h*)
BF18	max(0, *h* – 76) × max(0, *d*/*h* – 0.84)
BF19	max(0, *h* – 76) × max(0, 0.84 – *d*/*h*)
BF20	max(0, *h* – 76) × max(0, 2 – 100ρ)
BF21	max(0, *h* – 76) × max(0, 2.5 – 100ρ)
BF22	max(0, *b* – 70) × max(0, 15 – *L*/*h*)

$$\begin{aligned} N_u(kN) &= 103.7 - 7.533\times BF1 - 5.337\times BF2 + 0.117\times BF3 - 0.192\times BF4 \\ &+0.578\times BF5 - 0.052\times BF6 + 31.147\times BF7 + 12.537\times BF8 - 0.39\times BF9 \\ &-0.017\times BF10 + 38.11\times BF11 + 106.35\times BF12 - 3.85\times BF13 + 11.4\times BF14 \\ &-17.156\times BF15 - 32.7\times BF16 + 6.1\times BF17 + 95.82\times BF18 - 113\times BF19 + \\ &3.7\times BF20 - 3.059\times BF21 - 0.237\times BF22 \end{aligned} \tag{8}$$

4. Conclusions

This paper demonstrates the viability of using MARS for nonlinear structural modeling involving a multitude of design variables. Major findings obtained in this research include:

• MARS is capable of capturing the nonlinear structural relationships involving a multitude of variables with interaction among each other without making any specific assumption about the underlying functional relationship between the input variables and the response.

• The MARS technique is able to provide the relative importance of the input variables. Since it

explicitly defines the intervals for the input variables, the developed MARS models enables structural engineers to have better insights and understanding of where significant changes in the data may occur.

• The developed MARS model gives predictions that are just as accurate as other soft computing techniques. Nevertheless, with regard to the developed model interpretability, MARS outperforms other soft computing techniques.

It should be noted that since the built MARS models make predictions based on the knot values and the basis functions, thus interpolations between the knots of design input variables are more accurate and reliable than extrapolations. Consequently, it is not recommended that the model be applied for values of input parameters beyond the specific ranges in this study.

References

Alacali, S.N., Akbas, B. and Doran, B. (2011), "Prediction of lateral confinement coefficient in reinforced concrete columns using neural network simulation", *Appl. Soft Comput.*, **11**(2), 2645-2655.

Alavi, A.H. and Gandomi, A.H. (2011), "Prediction of principal ground-motion parameters using a hybrid method coupling artificial neural networks and simulated annealing", *Comput. Struct.*, **89**(23-24), 2176-2194.

Arafa, M., Alqedra, M. and Najjar, H.A. (2011), "Neural network models for predicting shear strength of reinforced normal and high-strength concrete deep beams", *J. Appl. Sci.*, **11**(2), 266-274.

Attoh Okine, N.O., Cooger, K. and Mensah, S. (2009), "Multivariate Adaptive Regression (MARS) and Hinged Hyperplanes (HHP) for Doweled Pavement Performance Modeling", *Constr. Build. Mater.*, **23**, 3020-3023.

Caglar, N. (2009), "Neural network based approach for determining the shear strength of circular reinforced concrete columns", *Constr. Build. Mater.*, **23**(10), 3225-3232.

Chua, C.G. (2001), "Prediction of the behavior of braced excavation systems using Bayesian neural networks", Master Thesis, Nanyang Technological University, Singapore.

Chua, C.G. and Goh, A.T.C. (2003), "A hybrid Bayesian back-propagation neural network approach to multivariate modeling", *Int. J. Numer. Anal. Meter.*, **27**, 651-667.

Chuang, P.H., Goh, A.T.C. and Wu, X. (1998), "Modeling the capacity of pin-ended slender reinforced concrete columns using neural networks", *J. Struct. Eng.*, **124**(7), 830-838.

Friedman, J.H. (1991), "Multivariate adaptive regression splines", *Ann. Stat.*, **19**, 1-141.

Gandomi, A.H., Alavi, A.H., Kazemi, S., Alinia, M.M. (2009). "Behavior appraisal of steel semi-rigid joints using linear genetic programming", *J. Constr. Steel Res.*, **65**, 1738-1750.

Gandomi, A.H., Yang, X.S., Talatahari, S., Alavi, A.H., (2013), *Metaheuristic Applications in Structures and Infrastructures*, Elsevier, Waltham, MA, USA.

Goh, A.T.C. (1995), "Neural networks to predict shear strength of deep beams", *ACI Struct. J.*, **92**(1), 28-32.

Goh, A.T.C. and Chua, C.G. (2004), "Nonlinear modeling with confidence estimation using Bayesian neural networks", *Elect. J. Struct. Eng.*, **1**, 108-118.

Goh, A.T.C. and Zhang, W.G. (2014), "An improvement to MLR model for predicting liquefaction-induced lateral spread using multivariate adaptive regression splines", *Eng. Geol.*, **170**, 1-10.

Gulec, C.K. (2009), "Performance-based assessment and design of squat reinforced concrete shear walls", Ph.D. Thesis, the State University of New York at Buffalo.

Hastie, T., Tibshirani, R. and Friedman, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd Edition, Springer.

Jekabsons, G. (2011), *ARESLab: Adaptive Regression Splines toolbox for Matlab / Octave*, Available at http://www.cs.rtu.lv/jekabsons/

Jenkins, W.M. (2006), "Neural network weight training by mutation", *Comput. Struct.*, **84**(31-32), 2107-

2112.
Lashkari, A. (2012), "Prediction of the shaft resistance of nondisplacement piles in sand", *Int. J. Numer. Anal. Meter.* **37**, 904-931.
Mackay, D.J.C. (1991), "Bayesian methods for adaptive models", Ph.D. Thesis, California Institute of Technology.
Mirzahosseini, M., Aghaeifar, A., Alavi, A., Gandomi, A. and Seyednour, R. (2011), "Permanent deformation analysis of asphalt mixtures using soft computing techniques", *Expert Syst. Appl.*, **38**(5), 6081-6100.
Neal, R.M. (1992), "Bayesian training of back-propagation networks by the hybrid Monte Carlo method", Technical report CRG-TG-92-1, Department of Computer Science, University of Toronto, Canada.
Oreta, A.W.C. and Kawashima, K. (2003), "Neural network modeling of confined compressive strength and strain of circular concrete columns", *J. Struct. Eng.*, **129**(4), 554-561.
Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986), "Learning internal representations by error propagation", *Parallel Distributed Processing*, Eds. D.E. Rumelhart & J.L. McClelland, MIT Press, Cambridge, MA.
Samui, P. (2011), "Determination of ultimate capacity of driven piles in cohesionless soil: a multivariate adaptive regression spline approach", *Int. J. Numer. Anal. Meter.*, **36**, 1434-1439.
Samui, P. and Karup, P. (2011), "Multivariate adaptive regression spline and least square support vector machine for prediction of undrained shear strength of clay", *IJAMC*, **3**(2), 33-42.
Samui, P., Das, S. and Kim, D. (2011), "Uplift capacity of suction caisson in clay using multivariate adaptive regression spline", *Ocean Eng.*, **38**, 2123-2127.
Sanad, A. and Saka, M.P. (2001), "Prediction of ultimate shear strength of reinforced-concrete deep beams using neural networks", *J. Struct. Eng.*, **127**(7), 818-828.
Tsai, H.C. (2010), "Hybrid high order neural networks", *Appl. Soft Comput.*, **9**, 874-881.
Tsai, H.C. (2011), "Using weighted genetic programming to program squat wall strengths and tune associated formulas", *Eng. Appl. Artif. Intel.*, **24**, 526-533.
Yang, K.H., Ashour, A.F., Song, J.K. and Lee, E.T. (2008), "Neural network modeling of RC deep beam shear strength", *Struct. Build.*, **161**(1), 29-39.
Zarnani, S., El-Emam, M. and Bathurst, R.J. (2011), "Comparison of numerical and analytical solutions for reinforced soil wall shaking table tests", *Geomech. Eng.*, **3**(4), 291-321.
Zhang, W. G. and Goh, A. T. C. (2013), "Multivariate adaptive regression splines for analysis of geotechnical engineering systems", *Comput. Geotech.*, **48**, 82-95.
Zhang, W.G. and Goh, A.T.C. (2014), "Multivariate adaptive regression splines model for reliability assessment of serviceability limit state of twin caverns", *Geomech. Eng.*, **7**(4), 431-458.
Zhang, W.G., Goh, A.T.C., Zhang, Y.M., Chen, Y.M. and Xiao, Y. (2015), "Assessment of soil liquefaction based on capacity energy concept and multivariate adaptive regression splines", *Eng. Geol.*, **188**, 29-37.

CC