



Combining machine learning models via adaptive ensemble weighting for prediction of shear capacity of reinforced-concrete deep beams

Doddy Prayogo¹ · Min-Yuan Cheng² · Yu-Wei Wu² · Duc-Hoc Tran³

Received: 26 December 2018 / Accepted: 16 April 2019 / Published online: 30 April 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract

This study presents a novel artificial intelligence (AI) technique based on two support vector machine (SVM) models and symbiotic organisms search (SOS) algorithm, called “optimized support vector machines with adaptive ensemble weighting” (OSVM-AEW), to predict the shear capacity of reinforced-concrete (RC) deep beams. This ensemble learning-based system combines two supervised learning models—the support vector machine (SVM) and least-squares support vector machine (LS-SVM)—with the SOS optimization algorithm as the optimizer. In OSVM-AEW, SOS is integrated to simultaneously select the optimal parameters of SVM and LS-SVM, and control the coordination process of the learning outputs. Experimental results show that OSVM-AEW achieves the greatest evaluation criteria for coefficient of correlation (0.9620), coefficient of determination (0.9254), mean absolute error (0.3854 MPa), mean absolute percentage error (7.68%), and root-mean-squared error (0.5265 MPa). This paper demonstrates the successful application of OSVM-AEW as an efficient tool for helping structural engineers in the RC deep beams design process.

Keywords Shear strength · RC deep beams · Ensemble model · Symbiotic organisms search · Support vector machine

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s00366-019-00753-w>) contains supplementary material, which is available to authorized users.

✉ Doddy Prayogo
prayogo@petra.ac.id
Min-Yuan Cheng
myc@mail.ntust.edu.tw
Yu-Wei Wu
d9305503@mail.ntust.edu.tw
Duc-Hoc Tran
tdhoc@hcmut.edu.vn

- ¹ Department of Civil Engineering, Petra Christian University, Jalan Siwalankerto 121-131, Surabaya 60236, Indonesia
- ² Department of Civil and Construction Engineering, National Taiwan University of Science and Technology, #43, Sec. 4, Keelung Rd, Taipei 106, Taiwan, ROC
- ³ Department of Construction Engineering and Management, Ho Chi Minh City University of Technology, Vietnam National University Ho Chi Minh, City (VNU-HCM), 268 Ly Thuong Kiet St., Dist. 10, Ho Chi Minh City, Viet Nam

1 Introduction

As structural members, reinforced-concrete (RC) deep beams have often been used as load distribution elements such as in folded plate construction, foundation walls, transfer girders, and pile caps in tall buildings. Despite their usefulness and popularity, RC deep beams are difficult to design, because various parameters non-linearly affect their behavior and shear strength. Notably, shear stress is a dominant failure mode of RC deep beams that tends to result in sudden, severe collapse and human loss. For decades, quantitative studies have explored shear strength and analyzed the behavior of RC deep beams, including the strut-and-tie model [1, 2], mechanism analysis using the upper bound theorem of plasticity theory and finite-element analyses [3]. However, these design procedures are linear; thus, they often provide estimated values that are not close to the real strength of RC deep beams [4].

In general, a variety of bodies publish design code procedures that seek to calculate the ultimate shear strength of RC deep beams; these bodies include the American Concrete Institute (ACI) [5], the Construction Institute Research Information Association (CIRIA Guide 2) [6], and the Canadian

Standard Association (CSA) [7]. However, these values, with respect to experimental tests, are at best conservative and at worst subpar. The following standard performance by the design code can be attributed to the large number of parameters and the non-linear relationships between the ultimate shear strength of the RC deep beam and the corresponding parameters. This ultimately leads to an inherent difficulty in creating a model that approximates mathematical shear strength.

In the past decades, artificial intelligence (AI) has been the subject of extensive research in the civil engineering field [8–14]. It has been reported that AI has successfully simulated the human inference process and, therefore, has become a powerful prediction technique. This enables civil engineers to predict the structural performance of concrete members. AI techniques have proven to be capable of capturing the complicated non-linear relationship between simply supported RC deep beams with its influencing parameters and providing highly accurate strength estimations.

Support vector machine (SVM) is a widely used machine learning technique that was proposed by Vapnik in 1995 [15]. As one of the most robust methods in the field of machine learning, the SVM algorithm belongs to the group of supervised models. Compared with other similar methods, SVM offers a range of beneficial characteristics, such as its above-average generalization performance, giving it the ability to produce top-quality decision boundaries based on a relatively small training data point subset. Furthermore, SVM offers the ability to model both non-linear and complex relations. It is, perhaps, for these reasons that SVM has shown effective levels of performance in a variety of real-world challenges and problems, even when applied to a variety of situations and applications.

Least-squares support vector machine (LS-SVM) is an extension of SVM that contains several advanced features demonstrated by its fast computation and generalized capacity [16]. LS-SVM has been used to tackle many complex and non-linear problems in engineering [17]. As part of the training process, LS-SVM uses a least-squares cost function method to derive a linear set of equations that exist in dual space. As such, the solution is derived using iterative methods, such as conjugate gradient, to efficiently solve linear equations.

The prevailing trend in AI techniques combines the merits of multiple models, the so-called “ensemble model,” to create a new ensemble model with significantly higher performance. Typically, an ensemble model can achieve superior performance when both the accuracy of its individual learners and the error diversity among the members are high. Therefore, compared with the method of using a single machine learner with a different subset of training data, or using a single machine learner with different training parameters, the method of using various machine learners is

likely to create strong ensemble models with high generalizability, because different machine learners may efficiently avoid making the same error. Furthermore, the comprehensive training data can be fully employed in the training process, which is essential in the event of limited training data. Chou and Pham [18] found that an ensemble model combining two or more strong machine learners can markedly outperform individual learners. By combining multiple machine learners, problems with weak predictors can potentially be mitigated [19] and the variance of the results can be reduced, because the aggregate results of several learners can be less noisy than the single result of one learner. Another study conducted by Acar and Rais-Rohani [20] revealed that an ensemble of prediction models with optimized weighting coefficients always outperforms its stand-alone members.

Studies suggest that the ensemble models have been demonstrated to be superior to single AI models when employed to solve various civil engineering problems such as construction cost and schedule success [21], disputes prediction in public–private partnership projects [22], concrete compressive strength [18], and estimating the peak shear strength of discrete fiber-reinforced soils [23]. However, despite their success elsewhere, ensemble models have not yet been applied to RC deep beam-related problems. In addition, the ensemble weighting coefficients are set as average values [18], which is likely to diminish the ensemble model’s performance. Thus, the current study investigates the efficacy of combining two distinct machine learning techniques: SVM and LS-SVM.

To achieve the greatest success using the ensemble model of SVM and LS-SVM, the user must determine the appropriate parameter values for both ensemble members. A sub-optimal parameter setting may undermine model predictability, whereas an optimal parameter setting can increase the accuracy of the prediction model. Parameters from SVM that must be pre-specified include the regularization parameter (C) and RBF kernel parameter (γ_1). Meanwhile, the LS-SVM model parameters that must be pre-specified include the regularization parameter (γ_2) and RBF kernel parameter (σ^2). In addition, it is worth noting that identification of the most suitable parameter values is a challenging task and represents an optimization problem [24–28]. Therefore, fusing SVM and LS-SVM with an optimization algorithm may produce an effective resolution to the above-mentioned issue. In addition, adding the adaptive weighting coefficient will be likely to boost the ensemble performance instead of setting an average weighting value.

Inspired by the symbioses of living organisms, Cheng and Prayogo [29] developed a metaheuristic-based optimization algorithm known as “symbiotic organisms search” (SOS). The SOS algorithm is relatively flexible and easy to use, even for beginners, due in part to a small number of control parameters. The initial research has shown that SOS is a superior algorithm

as compared with others, especially in terms of identifying optimal solutions. Over the past years, SOS has been used to solve problems in different research fields [30–37]. In addition, SOS is a reliable tool when paired with other AI techniques [24, 25, 38, 39]. Thus, SOS can become a promising optimization tool when integrated with SVM and LS-SVM.

This study develops and investigates a new approach, called “optimized support vector machines with adaptive ensemble weighting” (OSVM-AEW), to predict the shear strength of RC deep beams. The OSVM-AEW’s output is created by dynamically aggregating prediction values of the SVM and LS-SVM models. The SOS optimization algorithm is integrated to autonomously identify the optimal parameter settings, ensuring that OSVM-AEW achieves the best performance. The benefits of this newly developed model include: (1) it can operate autonomously without AI knowledge and (2) it predicts shear strength more accurately than single models do.

2 Methodology

2.1 Support vector machine

First introduced by Vapnik [15], SVM is a binary classifier of the non-linear subset. As a supervised learning-based method, SVM is often useful in both regression and classification applications. In this study, SVM is used for regression applications. In formal terms, an SVM algorithm uses data points expressed as an (x_i, y_i) pair, where x_i represents the feature vector $(x_{i1}, x_{i2}, \dots, x_{ip})$, p represents the number of features, $i = 1, \dots, n$, and n represents how many training observations apply. For regression applications, y_i is scalar vector representing the output of the training data. Using each data point, the SVM algorithm works in high-dimensional vector spaces to construct linear separating hyperplanes.

The following equation represents the decision function of SVM:

$$y(x) = \omega^T \phi(x) + b, \tag{1}$$

where $x \in R^n, y \in R$, and $\phi(x) : R^n \rightarrow R^{nh}$ is the mapping to the high-dimensional feature space and b is the bias term.

For regression problem, the R function is formulated using a constrained optimization, as shown in Eq. (2), while c denotes the regularization constant, which the user must optimize, and gives the relative weight of the second part as compared with the first part; ϵ_i is the training data error:

$$\text{Minimize: } R(\omega, \epsilon, b) = \frac{1}{2} \omega^2 + c \sum_{i=1}^n \epsilon_i^2, \tag{2}$$

$$\text{Subjected to: } y_i = \omega^T \phi(x_i) + b + \epsilon_i, \quad i = 1, \dots, n.$$

Using each data point, the SVM algorithm for regression works in high-dimensional vector spaces to construct linear separating hyperplanes. The best hyperplane can be

identified and defined as the hyperplane with the maximum distance to its training errors (ϵ). Due to the uses of training error ϵ_i , SVM for regression starts to penalize for errors that are greater than ϵ in magnitude. In addition, using kernel function $\phi(x)$, a non-linear enlarging of the original space can result from the higher dimension of space.

However, the objective function from Eq. (2) can be unsolvable when ω is infinite. Thus, the Lagrange multiplier optimal programming method is used to handle this task, as shown in following equation:

$$L(\omega, b, \epsilon, \xi) = R(\omega, \epsilon, b) - \sum_{i=1}^n \xi_i (\omega^T \phi(x_i) + b + \epsilon_i - y_i), \tag{3}$$

where ξ_k is the Lagrange multiplier. Following are the given conditions for optimality:

$$\begin{cases} \frac{\partial L}{\partial \omega} = 0 \rightarrow \omega = \sum_{k=1}^N \alpha_k \phi(x_k) \\ \frac{\partial L}{\partial b} = 0 \rightarrow \sum_{k=1}^N \alpha_k = 0 \\ \frac{\partial L}{\partial e_k} = 0 \rightarrow \alpha_k = \gamma e_k, \quad k = 1, \dots, N \\ \frac{\partial L}{\partial \alpha_k} = 0 \rightarrow \omega^T \phi(x_k) + b + e_k - y_k = 0, \quad k = 1, \dots, N \end{cases} \tag{4}$$

After eliminating ϵ and ω , the linear system is obtained as follows:

$$\begin{bmatrix} 0 & 1_v^T \\ 1_v & \omega + 1/\gamma \end{bmatrix} \begin{bmatrix} b \\ \xi \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \tag{5}$$

where $y = y_1, \dots, y_n$, $1_v = [1; \dots; 1]$, and $\xi = [\xi_1; \dots; \xi_n]$. The kernel function is applied as follows:

$$\omega = \phi(x_i)^T \phi(x_k) = K(x_i, x_k). \tag{6}$$

Finally, the SVM model for regression is presented as Eq. (7):

$$y(x) = \sum_{i=1}^n \xi_i K(x_i, x_k) + b, \tag{7}$$

$$K(x_k, x_l) = \exp\left(-\frac{x_k - x_l^2}{2\gamma^2}\right), \tag{8}$$

where ξ_k and b denote the solution to the linear system and γ is the kernel function parameter. This study uses the radial basis function (RBF) as the kernel function, as shown in Eq. (8).

2.2 Least-squares support vector machine (LS-SVM)

First introduced by Suykens et al. [16], the LS-SVM is a variant of the SVM. The R function is formulated using a constrained optimization for the function estimation problem, as shown in Eq. (9), while γ represents the regularization constant, which the user must optimize, and gives the relative weight of the second part as compared with the first part; ε_i is the training data error:

$$\begin{aligned} \text{Minimize:} \quad R(\omega, \varepsilon, b) &= \frac{1}{2} \|\omega\|^2 + \frac{1}{2} \gamma \sum_{i=1}^n \varepsilon_i^2, \\ \text{Subjected to: } y_i &= \omega^T \phi(x_i) + b + \varepsilon_i, \quad i = 1, \dots, n. \end{aligned} \quad (9)$$

It is worth noting that Eq. (9) is slightly similar to Eq. (2) from SVM's decision function. As can be seen, the objective function R represents the sum of the squared fitting error and a regularization term similar to the standard procedure in training the feed-forward neural network and is related to a ridge regression. Nevertheless, this objective function can be unsolvable when ω is infinite. Thus, the Lagrange multiplier optimal programming method is used to handle this task, as shown in following equation:

$$L(\omega, b, \varepsilon, \xi) = R(\omega, \varepsilon, b) - \sum_{i=1}^n \xi_i (\omega^T \phi(x_i) + b + \varepsilon_i - y_i), \quad (10)$$

Using a similar approach with the SVM described in previous section, the LS-SVM model for regression is presented as Eq. (11):

$$y(x) = \sum_{i=1}^n \xi_i K(x_i, x_k) + b, \quad (11)$$

$$K(x_k, x_l) = \exp\left(-\frac{\|x_k - x_l\|^2}{2\sigma^2}\right), \quad (12)$$

where ξ_k and b denote the solution to the linear system and σ^2 is the kernel function parameter. This study uses the radial basis function (RBF) as the kernel function, as shown in Eq. (12). Hence, to attain success in establishing the LS-SVM model, two tuning parameters (γ , σ^2) must be determined. The current study, thus, employs SOS to identify the optimal values of γ and σ^2 .

2.3 Symbiotic organisms search (SOS)

Initially proposed by Cheng and Prayogo [29] as a population-based optimization algorithm, the SOS algorithm is now becoming widely used in solving many multidimensional optimization problems [40]. For the optimization routine, SOS utilizes three types of symbioses: mutualism,

commensalism, and parasitism. The SOS algorithm uses organisms to represent potential solutions to the problem itself, with the organism's fitness referring to the associated solution's quality. Figure 1 shows the pseudo-code of SOS.

During the mutualism phase, new potential improved organisms are generated by the mutualism symbiosis between two organisms, which benefits both sides. To improve the candidate organisms (new_ X_i and new_ X_j), the current organisms (X_i and X_j) simulate a mutualism interaction as modeled in the following equations:

$$\text{new_}X_i = X_i + \text{rand}(0, 1) \times (X_{\text{best}} - \text{mutual_vector} \times \text{BF}_1), \quad (13)$$

$$\text{new_}X_j = X_j + \text{rand}(0, 1) \times (X_{\text{best}} - \text{mutual_vector} \times \text{BF}_2), \quad (14)$$

$$\text{mutual_vector} = \text{mean}(X_i + X_j), \quad (15)$$

where mutual_vector represents the mutualism relationship between current organisms and BF_1 and BF_2 are random values of either 0 or 1, illustrating each interacting organism's benefit level.

During the commensalism phase, a new potential improved organism is generated by the commensalism symbiosis between two organisms, which benefits only one side. To improve the candidate organism (X_i), the current organisms (X_i and X_j) simulate a mutualism interaction as modeled in the following equation:

$$\text{new_}X_i = X_i + \text{rand}(-1, 1) \times (X_{\text{best}} - X_j). \quad (16)$$

```

1: Initialization
2: Generate the initial population of organisms,  $X$ , and maximum iteration
3: Set Cycle = 0
4: Compute the fitness,  $fit$ , of the population
5: Identify the best organism of the initial population,  $X_{\text{best}}$ 
6: REPEAT
7:   Mutualism phase
8:   Select an organism  $X_j$  randomly to interact with  $X_i$ 
9:   Produce new organisms from the mutualism operator
10:  Evaluate the fitness value  $fit$ 
11:  Apply greedy selection process
12:  Commensalism phase
13:  Select an organism  $X_j$  randomly to interact with  $X_i$ 
14:  Produce a new organism from the commensalism operator
15:  Evaluate the fitness value  $fit$ 
16:  Apply greedy selection process
17:  Parasitism phase
18:  Select an organism  $X_j$  randomly to interact with  $X_i$ 
19:  Produce a  $parasite\_vector$  based on  $X_j$ 
20:  Evaluate the fitness value of  $parasite\_vector$ 
21:  If ( $parasite\_vector$  is better than  $X_j$ ) then
22:    Replace  $X_j$  with  $parasite\_vector$ 
23:  End
24:  Update the best solution
25:  Cycle = Cycle + 1
26: UNTIL Cycle = maximum iteration
27: End

```

Fig. 1 SOS pseudo-code

During the parasitism phase, a new potential improved organism is generated by the parasitism symbiosis between two organisms, which benefits one side and is harmful for the other side. In this circumstance, organism X_i will generate a parasite_vector to threaten the existence of X_j . The parasite_vector is generated using the following equation:

$$\text{Parasite_vector} = F \times X_i + (1-F) \times [\text{rand}(0, 1) \times (\text{up} - \text{low}) + \text{low}], \quad (17)$$

where F and $(1-F)$ represent the binary random generator and its inverse, respectively; up and low are the search space's upper and lower boundaries, respectively.

2.4 Optimized support vector machine models with adaptive ensemble weighting (OSVM-AEW)

In machine learning, identification of the most suitable tuning parameter values is a critical issue and is recognized as an optimization problem [16, 29, 41]. For the ensemble model, besides having to determine the optimal parameter settings for individual models, users must concurrently find the combination coefficient values for individual models involved in the ensemble model. This is a high-dimension optimization problem and often beyond human ability. Hence, to maximize efficient performance, the current study uses the SOS search engine to optimize tuning parameters in construction of the developed OSVM-AEW. In multiple attempts to increase the AI-based inference model's prediction accuracy, studies focusing on a combination of an ensemble model and a metaheuristic algorithm are surprisingly limited.

This section describes the operation of the proposed OSVM-AEW for estimating the shear of RC deep beams in detail, as shown in Fig. 2. The proposed OSVM-AEW was built in the MATLAB R2017b environment.

2.5 Data process

The range [0, 1] is used to normalize the training data using Eq. (18). This prevents the formation of numerical difficulties along with the problem of larger range numeric attributes dominating smaller range attributes. Next, the training data are randomly divided into ten distinct folds. While nine folds are used to construct an estimation model, the tenth fold is sequenced through validation tasks:

$$x_i^n = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}, \quad (18)$$

where x_i denotes any data point, x_{\min} denotes the minimum value of the entire data set, x_{\max} represents maximum value of the entire data set, and x_i^n denotes the normalized value of the data point.

2.6 Parameter initialization

Before commencement of the searching loop within pre-defined ranges, six tuning parameters (C , γ_1 , γ_2 , σ^2 , α , and β) must be generated as the initial values. Table 1 presents the settings for each parameter of the experiment in this study, as suggested by the previous studies [25, 28].

2.7 SVM and LS-SVM prediction models

During this stage, SVM and LS-SVM carry out their training process using parameter values (C , γ_1 , γ_2 , and σ^2) provided by the SOS search engine and then test the validation data. It is worth noting that these parameter values seek to achieve the highest prediction accuracy for the OSVM-AEW as opposed to individual models.

2.8 Ensemble learning system

The proposed ensemble model's prediction value is the sum of individual models' prediction values multiplied by their own ensemble weighting coefficients (α and β), as shown in Eq. (19). SOS searching produces the ensemble weighting coefficients (α and β). The literature reveals that these coefficients are subjectively set as average values [18], which is likely to diminish the ensemble model performance:

$$P_i^{\text{OSVM-EL}} = \alpha \times P_i^{\text{SVM}} + \beta \times P_i^{\text{LS-SVM}}, \quad (19)$$

where $P_i^{\text{OSVM-EL}}$, P_i^{SVM} , and $P_i^{\text{LS-SVM}}$ indicate the prediction values of OSVM-AEW, SVM, and LS-SVM, respectively, and α and β are the ensemble weighting coefficients of SVM and LS-SVM, respectively.

2.9 Evaluate ensemble model using fitness function

Through quantitative studies, a metaheuristic algorithm has been combined with a machine learner to create a hybrid algorithm. Nevertheless, all attempts have a common problem; the training error is often not optimally minimized during the training process. On the other hand, this well fitting of the training set could simply reflect the model's complexity, which tends to face the over-fitting issue [42]. To combat the effects of over-fitting and to tune hyperparameters, one should consider validation of the training set during construction of the inference model. This study used k -fold cross-validation to partition the training data as commonly used in many studies [43, 44]. The number of folds is usually determined by the number of data points. In selecting the number of folds (k), one should ensure that the number of data points in the training subset and the validation subset contain sufficient variation and represent the same distribution. If a data set only contains 10 data points, then using tenfold cross-validation will result in only one data point in the validation subset. In this study, we used 67% as the proportion for training subset

Fig. 2 Structure of OSVM-AEW

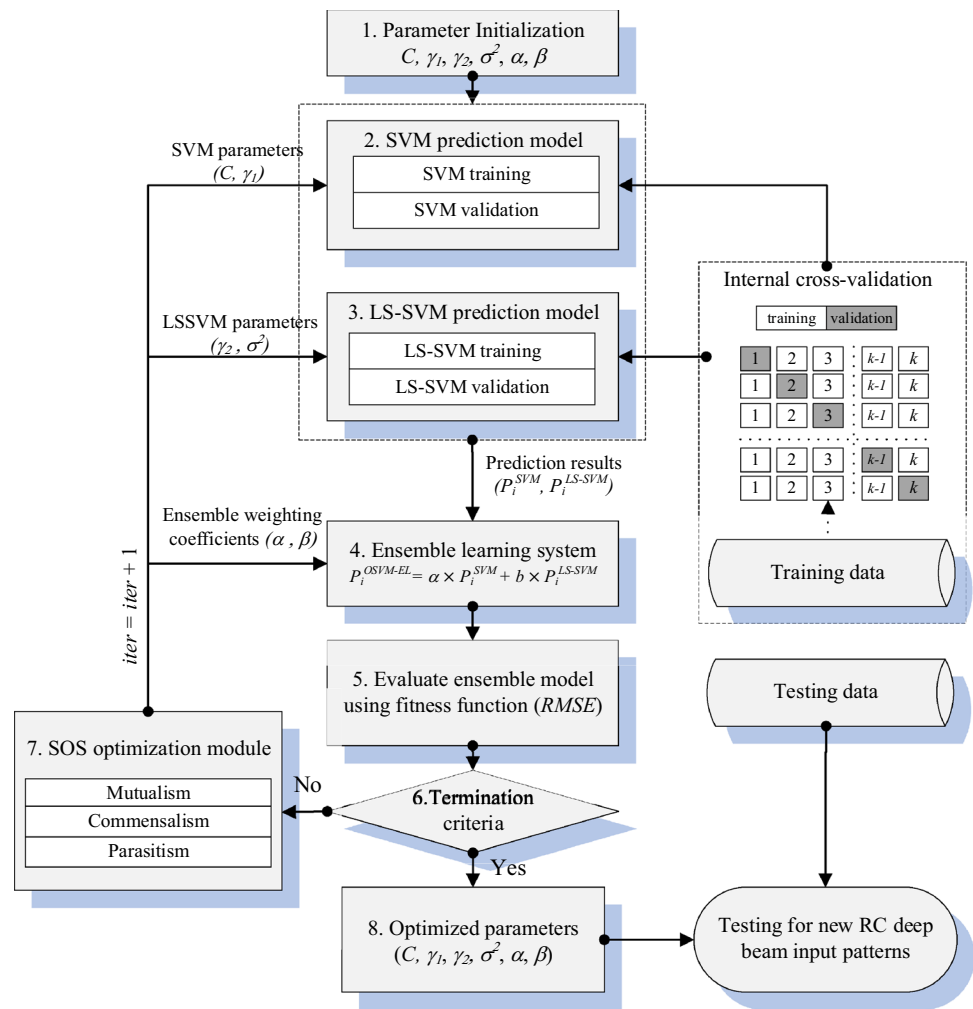


Table 1 Initial control parameter settings

Parameters	Notation	Lower bound	Upper bound	References
SVM regularization parameter	C	10^{-5}	10^5	[28]
SVM RBF kernel parameter	γ_1	10^{-5}	10^5	[28]
LS-SVM regularization parameter	γ_2	10^{-5}	10^5	[25, 28]
LS-SVM RBF kernel parameter	σ^2	10^{-5}	10^5	[25, 28]
Ensemble learning coefficients	α, β	0	1	

and 33% for the validation subset, which can be achieved by implementing the threefold cross-validation approach, to ensure that all subsets have the same distribution.

Because the present study uses threefold cross-validation, the ensemble model will be trained and validated three times for each set in the parameter optimizing process. As such, the objective function is arranged with the average validation error over the threefold. The prediction accuracy function for identifying the optimal tuning parameters is used as the SOS algorithm’s objective function and is shown as follows:

$$f = \text{RMSE}_{\text{validation}}, \tag{20}$$

where $\text{RMSE}_{\text{validation}}$ indicates the average validating error of threefolds. In Eq. (20), root-mean-square error (RMSE) is used as the estimation error. The RMSE is computed using the following equation:

$$\text{RMSE} = \sqrt{(1/n) \times \sum_{i=1}^n [p_i - y_i]^2}, \tag{21}$$

where p_i predicted value, y_i actual value, and n sample size.

2.10 SOS searching

For OSVM-AEW, the best tuning parameters, C , γ_1 , γ_2 , σ^2 , α , and β , are identified by comparing the qualities of the fitness function or the objective function. In this way, the tuning parameters have values that progressively improve through each loop. The smaller the fitness function’s value, the better the tuning parameter values. Through the greedy selection of the SOS algorithm, optimal tuning parameters, which provide the smallest fitness function, can be selected.

2.11 Stopping criterion

Once the stopping criterion that the user has designated has been met, the optimization process is stopped. In this study, the criterion to stop the model is the maximum number of iterations (max_iter).

2.12 Optimal parameters

Once the stop criterion has been met, the loop terminates. This indicates that the tuning parameters, which include C , γ_1 , γ_2 , σ^2 , α , and β , have reached an optimal set and are ready to predict a new and unseen input dataset.

3 Experimental results and discussion

3.1 Data set description

In this study, the data set comprised 214 deep beam test results conducted throughout the course of eight independent studies [2]. These studies included: 37 test patterns from Clark [45], 25 test patterns from Kong et al. [46], 52 test patterns from Smith and Vantsiotis [47], 12 test patterns from Anderson and Ramirez [48], 19 test patterns from Tan et al. [49], 53 test patterns from Oh and Shin [50], 4 test patterns from Aguilar et al. [51], and 12 test patterns from Quintero-Febres et al. [52]. The RC deep beams used in the experiments were simply supported; each test ran until failure. Figure 3 illustrates the details for an RC deep beam.

Following a review of the literature, eight essential factors affecting the shear strength of RC deep beams are selected [53]. The first seven influencing factors, used as input variables in the OSVM-AEW to estimate shear strength, are: effective depth (d); web width (b); main reinforcement ratio (ρ); concrete compressive strength (f_c); vertical shear reinforcement ratio (ρ_v); horizontal shear reinforcement ratio (ρ_h); and ratio of shear span to effective depth (a/d). The remaining factor is the ultimate shear strength of RC deep beam (V/bd). Table 2 and Fig. 4 present a statistical description of the input and output variables.

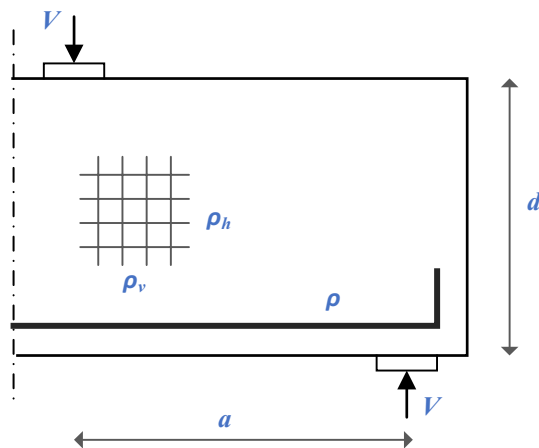


Fig. 3 Geometrical parameters of an RC deep beam

3.2 Performance indicators

The performance of OSVM-AEW was compared with that of other models using the coefficient of correlation (R), coefficient of determination (R^2), mean absolute percentage error (MAPE), mean absolute error (MAE), and root-mean-square error (RMSE). The coefficient R represents the correlation between the actual values and predicted values. The coefficient R^2 represents the level of explained variability between the actual values and predicted values. The closer the coefficients R and R^2 vary toward 1, the more similar the actual and predicted values become. Similarly, relatively low values for MAE, MAPE, and RMSE indicate a high level of accuracy in the values that the model predicts. Equations (22)–(26) show the mathematical formulation of the proposed performance evaluation criteria:

$$RMSE = \sqrt{(1/n) \times \sum_{i=1}^n [p_i - y_i]^2}, \tag{22}$$

$$MAE = (1/n) \times \sum_{i=1}^n (|p_i - y_i|), \tag{23}$$

$$MAPE = (1/n) \sum_{i=1}^n (|p_i - y_i|/y_i) \times 100, \tag{24}$$

$$R = \frac{n \sum y_i \times p_i - (\sum y_i)(\sum p_i)}{\sqrt{n(\sum y_i^2) - (\sum y_i)^2} \times \sqrt{n(\sum p_i^2) - (\sum p_i)^2}}, \tag{25}$$

where p_i predicted value, y_i actual value, and n sample size:

Table 2 Statistical description of any input and output variables used in this study

Variables	Unit	Min.	Max.	Mean	Std.
X_1 = effective depth (d)	mm	216	801	420.8	108.46
X_2 = web width (b)	mm	76	305	134.2	48.95
X_3 = Concrete compressive strength (f_c)	MPa	13.80	73.60	33.35	15.93
X_4 = Ratio of shear span to effective depth (a/d)	–	0.27	2.70	1.22	0.53
X_5 = Main reinforcement ratio (ρ)	%	0.52	4.08	1.81	0.70
X_6 = Horizontal shear reinforcement ratio (ρ_h)	%	0.00	2.47	0.32	0.45
X_7 = Vertical shear reinforcement ratio (ρ_v)	%	0.00	2.65	0.54	0.66
Y = Ultimate shear strength of RC beam (V/bd)	MPa	1.73	13.25	5.37	2.15

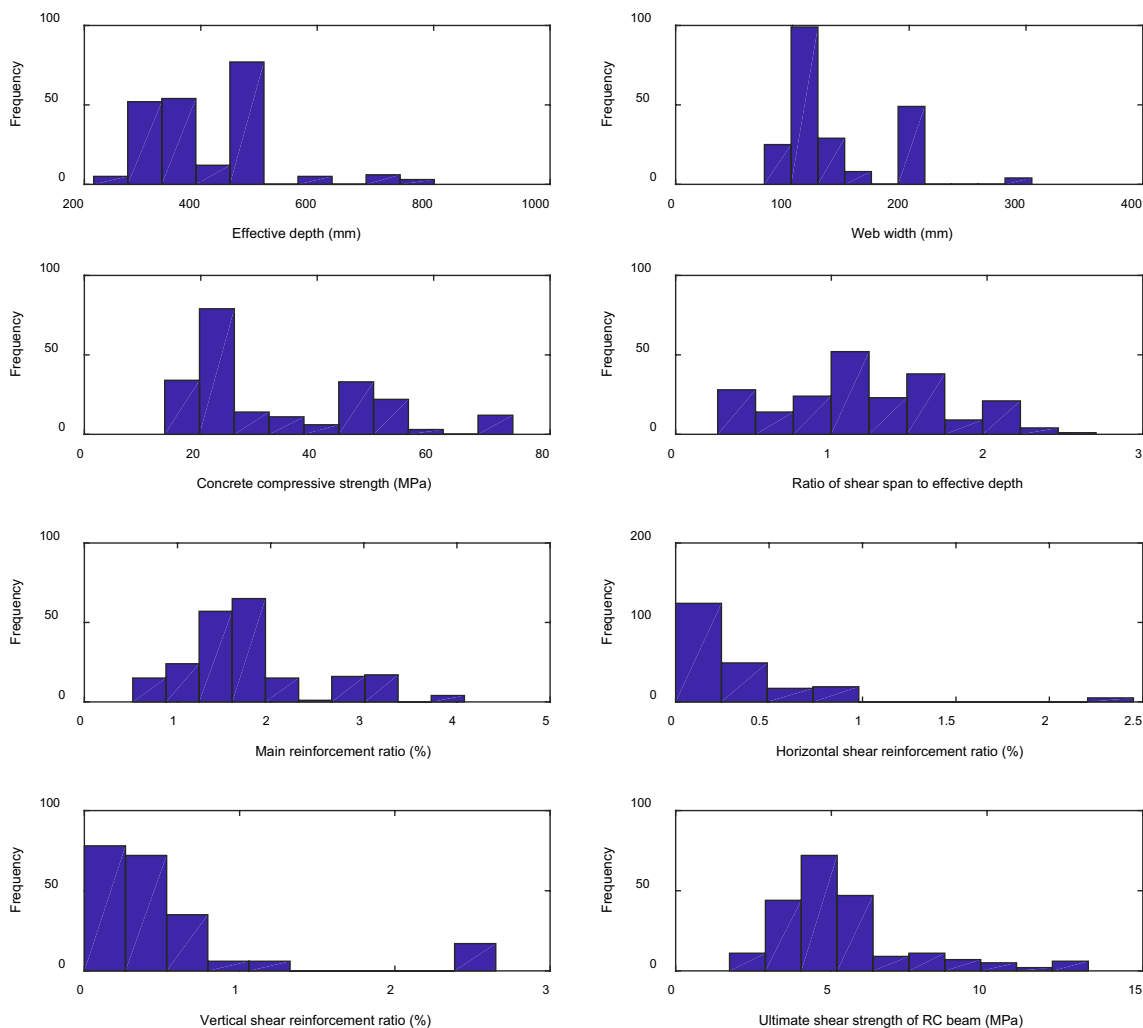


Fig. 4 Histogram of the input and output variables

$$R^2 = \left(\frac{n \sum y_i \times p_i - (\sum y_i)(\sum p_i)}{\sqrt{n(\sum y_i^2) - (\sum y_i)^2} \times \sqrt{n(\sum p_i^2) - (\sum p_i)^2}} \right)^2, \tag{26}$$

where p_i predicted value, y_i actual value, and n sample size.

3.3 Experimental results

In this study, the data set was randomly partitioned to training and testing. The complete training and testing data sets can be seen at Appendix. A training partition was used to build the prediction model. The testing partition was used to test the established trained model. A total of 171 data points were selected randomly as a training partition and the remaining were used for testing. The training partition was further divided into two sub-data sets. The training set is primarily used for model fitting. The validation set is used to validate the previous training performance and to fine-tune the trained model. A threefold cross-validation approach is used in this study to mitigate any bias that might have occurred in the model training due to the randomness of selecting training and validation subsets [42].

3.4 Training phase and parameter selection

Through the cross-validation approach, the training data points were randomly grouped into three mutually exclusive subsets. During each iteration, one subset was used for the validation process, while the other two subsets were utilized in turn for the training process. In this way, the study ensured that each data set was used at least once during both the testing and training phases. As such, the average results of the three subsets represent a useful way of predicting the model’s overall performance.

The training results of the cross-validation approach and the convergence graphics of each fold’s RMSEs are depicted in Fig. 5. It can be seen that, through parameter optimization process by SOS, the average validation RMSE was improved in every iteration.

The tuning process of hyperparameters by SOS is illustrated in Fig. 6. The achieved values of α and β properly vary to yield the best collaboration between SVM and LS-SVM rather than follow the 50:50 assumption rule of Chou and Pham [18]. Moreover, the sum of α and β in each fold is very close to 1, though this constraint was not applied in the model construction process.

Finally, the optimal parameters obtained through the fine-tuning process are shown in Table 3. Figure 7 shows the training performance based on the obtained optimal

Fig. 5 Convergence graphics of RMSE_{validation} of each fold

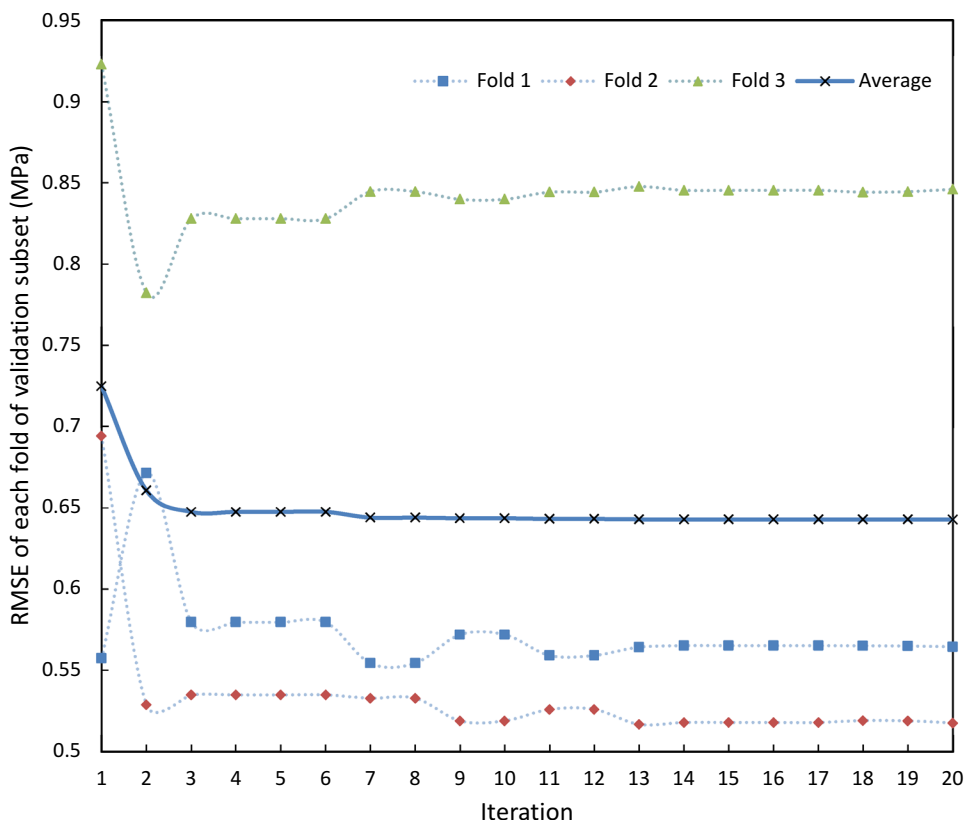


Fig. 6 Tuning process of hyperparameters and ensemble learning coefficients

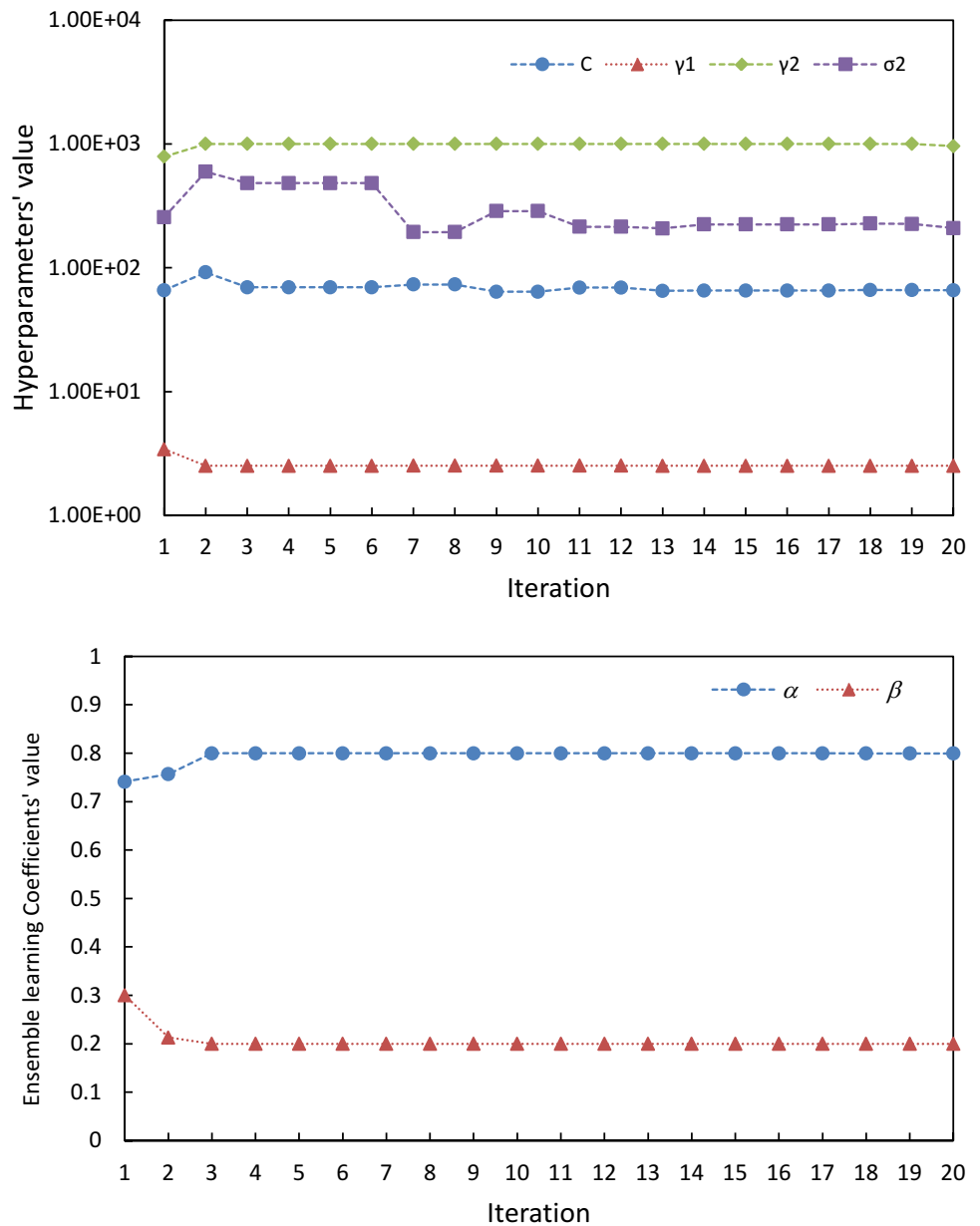


Table 3 Obtained optimal parameters of OSVM-AEW through training phase

Parameters	Obtained optimal value by SOS
SVM	
C	66.2800
γ_1	2.5119
LS-SVM	
γ_2	928.1491
σ^2	215.9683
Ensemble weighting coefficients	
α	0.7992
β	0.2001

parameters. According to the four performance indicators, OSVM-AEW produced high value of R and R^2 as well as low values of RMSE, MAPE, and MAE. Thus, these optimal parameters and model are ready to predict the new and unseen testing data set.

3.5 Testing phase and prediction results

Using the optimal parameters from Table 3 and the trained model, the testing result of OSVM-AEW is displayed in Fig. 8. It is observable that the OSVM-AEW earned relatively high values of R (≥ 0.95) for testing. This indicates that the OSVM-AEW is proficient at capturing the shear strength's underlying function. Because of the reaping of low MAPE values ($\leq 8\%$), the OSVM-AEW-estimated shear

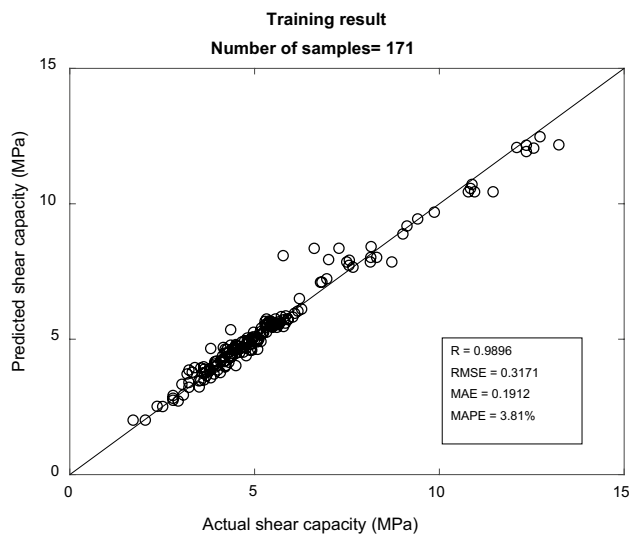


Fig. 7 Training performance and result of OSVM-AEW

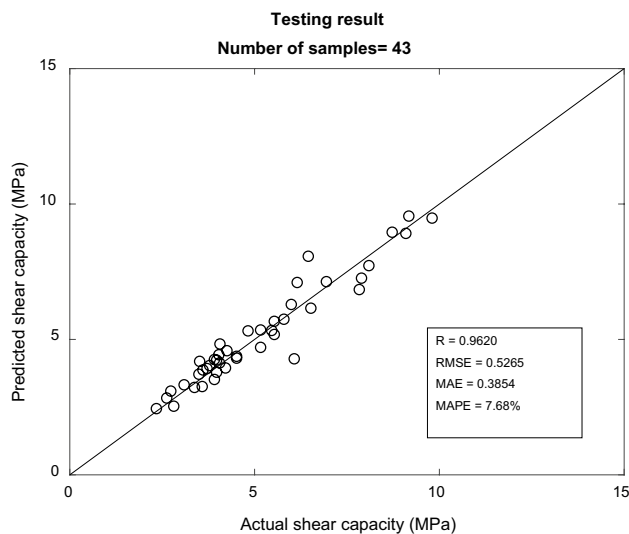


Fig. 8 Testing result of OSVM-AEW

strength is reliable and trustworthy. Thus, it is acceptable for the design of RC deep beams. Furthermore, the statistical results show that the proposed method can avoid the overfitting issue by attaining minor differences between performance evaluation values for the training and testing phases.

3.6 Benchmark with other AI models

To better evaluate the developed OSVM-AEW's capability, its performance was compared against those of other AI techniques, including: (1) the SVM model without any

parameter tuning, (2) the LS-SVM model without any parameter tuning, (3) the optimized SVM model via SOS optimization, (4) the optimized LS-SVM model via SOS optimization, (5) the ensemble model of multiple linear regression and regression tree (MLR-RegTree), and (6) the ensemble model of SVM and linear SVM (SVM-LSVM) without involving any parameter tuning. The first five AI techniques were established in the MATLAB R2017b environment. The SVM model was adopted from LIBSVM toolbox [54], whereas the LS-SVM model was adopted from LS-SVMlab toolbox [55]. Meanwhile, the MLR-RegTree utilized “fitlm” and “fitrtree” function from MATLAB toolbox. The ensemble of SVM and LS-SVM was constructed using the IBM SPSS Modeler 18.1 platform. The parameters of these AI models are set to default as can be seen in Table 4. Table 5 reveals the results of competitive models with respect to average values and standard deviations.

One can see that the proposed model achieves the most desirable values of RMSE (0.5265), MAE (0.3854), MAPE (7.68), R (0.9620), and R^2 (0.9254). According to Table 5, the obtained results suggest that the OSVM-AEW outperforms other comparative AI models. The finding can be summarized as follows:

- The OSVM-AEW outperforms the SVM and LS-SVM. This indicates that parameter tuning plays an important role in establishing the prediction model. SVM and LS-SVM used a default parameter setting, while the proposed model utilized SOS for parameter optimization.
- The OSVM-AEW shows better performance than the optimized SVM and LS-SVM models. It can be concluded that the proposed ensemble learning system improves the performance of the single model in addition to the parameter tuning.
- The ensemble weighting coefficients are still needed to be optimized. Through the SOS optimization, OSVM-AEW is able to find the optimal weighting that results in the superior performance of OSVM-AEW over SVM-LSVM.
- According to Tables 3 and 5, the ensemble weighting coefficients of SVM (α) and LS-SVM (β) are 0.7992 and 0.2001, respectively, indicating that the SVM results dominate the ensemble's prediction. The previous study conducted by Acar and Rais-Rohani [20] revealed that optimizing ensemble weighting coefficients based on mean-squared error of each member will yield the best accuracy in comparison with the other approaches. According to Table 5, the RMSE of optimized SVM is slightly better than those of optimized LS-SVM. Thus, it can be concluded that OSVM-

Table 4 Features and parameters of AI methods used in this study

AI methods	Features		Parameters	
	Parameter tuning	Single or ensemble?	Values/range	References
SVM	No	Single	$C = 1$ $\gamma_1 = 1/n$ $n = \text{Number of input variables}$	[56]
LS-SVM	No	Single	$\gamma_2 = 1$ $\sigma^2 = 1$	[16]
Optimized SVM	Yes, via SOS optimization	Single	$C = 10^{-5} - 10^5$ $\gamma_1 = 10^{-5} - 10^5$	[28]
Optimized LS-SVM	Yes, via SOS optimization	Single	$\gamma_2 = 10^{-5} - 10^5$ $\sigma^2 = 10^{-5} - 10^5$	[25, 28]
MLR-RegTree (fitlm-fitrree)	No	Ensemble, with fixed ensemble weight	$\alpha = 0.5$ $\beta = 0.5$	Default setting from MATLAB R2017b
SVM-LSVM	No	Ensemble, with fixed ensemble weight	$C = 10$ $\gamma_1 = 0.1$ epsilon = 0.1 $\alpha = 0.5$ $\beta = 0.5$	Default setting from IBM SPSS Modeler 18.1
OSVM-AEW	Yes, via SOS optimization	Ensemble, with adaptive ensemble weight via SOS optimization	$C = 10^{-5} - 10^5$ $\gamma_1 = 10^{-5} - 10^5$ $\gamma_2 = 10^{-5} - 10^5$ $\sigma^2 = 10^{-5} - 10^5$ $\alpha = 0 - 1$ $\beta = 0 - 1$	[25, 28]

Table 5 Result comparisons with other AI approaches

Models	Testing result					Obtained parameters
	RMSE (MPa)	MAE (MPa)	MAPE (%)	R	R ²	
SVM	1.2167	0.9040	16.51	0.8717	0.7598	$C = 1$ $\gamma_1 = 1/7$
LS-SVM	0.7360	0.5669	12.27	0.9271	0.8595	$\gamma_2 = 1$ $\sigma^2 = 1$
Optimized SVM	0.5508	0.4035	8.26	0.9586	0.9189	$C = 46.7062$ $\gamma_1 = 3.0161$
Optimized LS-SVM	0.5514	0.4084	8.09	0.9586	0.9189	$\gamma_2 = 248.0875$ $\sigma^2 = 3.5211$
MLR-RegTree	0.6799	0.5150	10.00	0.9636	0.9285	$\alpha = 0.5$ $\beta = 0.5$
SVM-LSVM	0.7832	0.5940	11.31	0.9156	0.8383	$C = 10$ $\gamma_1 = 0.1$ epsilon = 0.1 $\alpha = 0.5$ $\beta = 0.5$
OSVM-AEW	0.5265	0.3854	7.68	0.9620	0.9254	$C = 66.2800$ $\gamma_1 = 2.5119$ $\gamma_2 = 928.1491$ $\sigma^2 = 215.9683$ $\alpha = 0.7992$ $\beta = 0.2001$

Bold numbers indicate best results

Table 6 Benchmark with the results from literature

Models	MAE (MPa)		MAPE (%)		$V_{\text{actual}}/V_{\text{predicted}}$	
	Avg.	Std.	Avg.	Std.	Avg.	Std.
ACI 318-11 [5]	2.060	1.225	39.05	16.21	1.766	0.571
CSA [7]	1.632	0.825	34.56	18.91	1.637	0.574
GSA [53]	0.617	0.506	12.19	10.87	1.007	0.147
OSVM-AEW	0.3854	0.3047	7.68	6.00	1.0030	0.1054

Bold numbers indicate best results

AEW ensemble learning system favors the SVM model, because SVM has a better *RMSE* than LS-SVM.

3.7 Benchmark with literature results

An additional experiment was conducted to compare OSVM-AEW's performance with those of two approaches widely applied in current structural design projects and research, including ACI 318-11 [5] and CSA [7], and genetic-simulated annealing (GSA) [53]. Table 6 shows the average and standard deviation values of models in terms of MAE, MAPE, and the actual value-to-estimated value ratio. In terms of $V_{\text{actual}}/V_{\text{estimated}}$, the ACI code and CSA code tend to provide shear strength values that are considerably lower than actual shear strength values, with average values of 1.766 and 1.637, and standard deviation values of 0.571 and 0.574, respectively. The GSA performed well by providing an average value of close to 1 (desirable value). However, its standard deviation value was relatively high compared with that of OSVM-AEW; thus, it is likely to deliver estimated values that are drastically different from the actual values. Regarding MAE and MAPE, the proposed model achieved the most desirable values for both the average and standard deviation, which are remarkably smaller than those of other approaches.

The results show that both mathematical methods, including ACI code and CSA code, are insufficient to estimate the shear strength of RC deep beams. Interestingly, the proposed model is particularly accurate for RC deep beams with high shear strength capacity. In sum, the analysis results have amply demonstrated that OSVM-AEW is a promising new tool for the bridge engineer and capable of producing reliable estimation values for the shear strength of RC deep beams. The improvement in the estimation accuracy of shear strength implies benefits in terms of increased safety and cost savings in designing RC deep beams.

4 Conclusions

This study introduces a novel AI-inference model, called “optimized support vector machines with adaptive ensemble weighting” (OSVM-AEW), to predict the shear strength of RC deep beams. In OSVM-AEW, two individual prediction models, SVM and LS-SVM, are controlled by the SOS search engine, which uses an optimization process to provide the most appropriate set of parameter values (C , γ_1 , γ_2 , σ^2 , α , and β). Thus, the proposed model can run autonomously without the need for human intervention and domain knowledge while still achieving the best performance.

This study is apparently the first to investigate the efficacy of an ensemble model for estimating shear strength for RC deep beams. A simulation involving 214 test beams shows that OSVM-AEW achieves the most desirable evaluation criteria values in terms of RMSE (0.5265 MPa), MAE (0.3854 MPa), MAPE (7.68%), and R (0.9620). This strongly indicates that OSVM-AEW is the most accurate model in terms of estimating shear strength for RC deep beams. These results indicate a drastic reduction in RMSE, MAE, and MAPE compared with other AI competitors and literature approaches. In sum, these findings fully justify the use of SVM, LS-SVM, and SOS to establish OSVM-AEW. Improvement of the estimation accuracy of shear strength implies increased safety in the design of RC deep beams.

Acknowledgments The authors gratefully acknowledge that the present research is supported by The Ministry of Research, Technology, and Higher Education of the Republic of Indonesia (No: 123.58/D2.3/KP/2018) under the “World Class Professor” (WCP) Research Grant Scheme.

Appendix 1

See Tables 7 and 8.

Table 7 Training data set

Case number	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
1	305	102	20.8	1	1.94	0.68	1.25	5.74
2	305	102	21.7	1.21	1.94	0	0	4.79
3	305	102	19.5	1.21	1.94	0.91	0.24	4.93
4	305	102	19.2	1.21	1.94	0.23	0.42	4.15
5	305	102	19	1.21	1.94	0.45	0.42	4.22
6	305	102	21.8	1.21	1.94	0.68	0.42	4.82
7	305	102	16.2	1.21	1.94	0.23	0.63	4.2
8	305	102	20.7	1.5	1.94	0	0	3.72
9	305	102	19.2	1.5	1.94	0.45	0.31	3.33
10	305	102	16.5	1.5	1.94	0.45	0.56	4.02
11	305	102	18.3	1.5	1.94	0.68	0.56	4.1
12	305	102	19	1.5	1.94	0.91	0.56	4.41
13	305	102	18.5	1.5	1.94	0.45	0.63	4.13
14	305	102	19.2	1.5	1.94	0.45	0.77	4.9
15	597	76	24.6	0.43	0.62	0	2.45	4.94
16	343	76	21.2	0.74	1.09	0	2.45	6.29
17	597	76	18.6	0.43	0.62	0	0.86	4.94
18	597	76	19.2	0.43	0.62	0.61	0.61	4.58
19	390	203	24.6	2.34	3.1	0	0.38	2.81
20	390	203	23.4	1.95	3.1	0	0.37	3.52
21	390	203	25.4	1.95	3.1	0	0.37	3.24
22	390	203	24.9	1.95	3.1	0	0.73	4.23
23	390	203	24.1	1.56	2.07	0	0.69	4.09
24	390	203	13.9	1.56	2.07	0	0.34	2.38
25	390	203	45.2	1.56	3.1	0	0.34	5.35
26	390	203	44.7	1.56	3.1	0	0.34	5.49
27	390	203	47.6	1.56	3.1	0	0.34	5.41
28	500	130	49.1	0.5	1.56	0	0	9.88
29	500	130	49.1	2	1.56	0.43	0.22	3.65
30	500	120	50.7	0.5	1.29	0.47	0.13	8.17
31	500	120	73.6	1.25	1.29	0.47	0.13	5.64
32	370	150	22	1.42	2.79	0.1	0.28	4.52
33	370	150	22	1.42	2.79	0.1	0.28	4.27
34	370	150	22	1.42	2.79	0	0	3.98
35	375	150	32.4	0.81	2.04	0	0	8.32
36	380	100	50.3	1.57	4.08	0.15	0.38	6.97
37	380	100	50.3	0.82	4.08	0	0	12.11
38	463	110	58.8	0.27	1.23	0	0.48	13.25
39	463	110	57.3	0.27	1.23	0	0.48	12.37
40	463	110	44.1	1.08	1.23	0	0.48	5.5
41	463	110	46.8	1.08	1.23	0	0.48	5.69
42	425	203	41.4	2.15	2.67	0	2.65	5.67
43	425	203	42.7	2.15	2.67	0	2.65	5.92
44	305	102	20.5	1	1.94	0	0	4.48
45	305	102	20.5	1	1.94	0.68	0.28	5.49
46	305	102	19.1	1	1.94	0.91	0.63	5.2
47	305	102	22.1	1.21	1.94	0.23	0.24	4.74
48	305	102	19.8	1.21	1.94	0.91	0.42	4.67
49	305	102	20.4	1.21	1.94	0.23	0.77	5.11
50	305	102	19.2	1.21	1.94	0.68	0.77	4.98
51	305	102	20.6	1.21	1.94	0.91	0.77	5.34

Table 7 (continued)

Case number	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
52	305	102	22.7	1.5	1.94	0.68	0.18	4.21
53	305	102	18.5	1.5	1.94	0.68	0.77	4.91
54	305	102	21.2	1.5	1.94	0.91	0.77	5.13
55	724	76	22.6	0.35	0.52	2.45	0	5.02
56	597	76	21	0.43	0.62	2.45	0	4.98
57	390	203	24.6	1.95	3.1	0	0.37	3.05
58	390	203	23.2	1.95	3.1	0	0.73	3.8
59	390	203	25.6	1.56	2.07	0	0.34	3.51
60	390	203	29	1.56	2.07	0	0.34	3.61
61	390	203	25	1.56	2.07	0	0.69	3.8
62	390	203	13.8	1.56	2.07	0	0.34	2.53
63	390	203	26.1	1.17	1.63	0	0.46	4.51
64	390	203	25.9	1.17	1.63	0	0.61	3.94
65	390	203	23.1	1.17	1.63	0	1.22	3.94
66	500	130	23.7	0.85	1.56	0.43	0.12	4.37
67	500	130	23.7	0.85	1.56	0.43	0.34	5.5
68	500	130	49.1	0.5	1.56	0.43	0.34	10.9
69	500	130	49.1	1.25	1.56	0.43	0.12	5.34
70	500	130	49.1	1.25	1.56	0.43	0.34	6.19
71	500	130	49.1	2	1.56	0	0	1.73
72	500	120	50.7	0.5	1.29	0	0.13	5.79
73	500	120	50.7	0.85	1.29	0.47	0.37	6.23
74	500	120	50.7	1.25	1.29	0.23	0.13	4.34
75	500	120	73.6	0.5	1.29	0	0.13	7.3
76	500	120	73.6	1.25	1.29	0.23	0.13	5.17
77	500	130	23.7	1.25	1.56	0.43	0.12	3.92
78	500	130	23.7	1.25	1.56	0.43	0.12	3.19
79	500	130	49.1	0.5	1.56	0.43	0.12	10.8
80	718	305	32	1.27	1.4	0.13	0.31	5.18
81	801	305	28	1.14	1.25	0.06	0.31	5.26
82	801	305	28	1.14	1.25	0	0.1	5.23
83	380	100	50.3	1.43	4.08	0	0	7.68
84	463	110	51.6	0.27	1.23	0	0.48	12.37
85	425	203	32.3	2.15	2.67	0	2.65	4.27
86	305	102	18	1	1.94	0.45	0.28	4.77
87	305	102	16.1	1	1.94	0.68	0.28	4.54
88	305	102	21.1	1	1.94	0.91	0.28	5.92
89	305	102	21.7	1	1.94	0.23	0.63	5.61
90	305	102	20.3	1	1.94	0.68	0.63	5.53
91	305	102	19.2	1	1.94	0.45	1.25	5.55
92	305	102	19.9	1	1.94	0.91	1.25	5.4
93	305	102	20.1	1.21	1.94	0.45	0.24	4.62
94	305	102	20.8	1.21	1.94	0.68	0.24	4.51
95	305	102	17.1	1.21	1.94	0.23	1.25	4.93
96	305	102	21.9	1.5	1.94	0.45	0.18	3.97
97	305	102	19.3	1.5	1.94	0.45	0.31	3.71
98	305	102	16.1	2.08	1.94	0.23	0.42	2.81
99	470	76	19.9	0.54	0.79	0	0.86	6.05
100	343	76	22.8	0.74	1.09	0	0.86	5.37
101	343	76	21.9	0.74	1.09	2.45	0	6.1
102	724	76	22	0.35	0.52	0.86	0	4.4

Table 7 (continued)

Case number	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
103	470	76	20.1	0.54	0.79	0.86	0	5.07
104	216	76	22.6	1.18	1.73	0.61	0.61	4.75
105	390	203	23.4	2.34	3.1	0	0.38	2.81
106	390	203	42.1	1.95	3.1	0	0.37	4.79
107	390	203	24	1.17	1.63	0	0.61	3.66
108	390	203	24.8	1.17	1.63	0	0.61	4.22
109	390	203	28.2	1.17	2.44	0	0.92	4.99
110	500	130	23.7	0.85	1.56	0.43	0.22	5.8
111	500	130	49.1	0.5	1.56	0.43	0.12	10.97
112	500	130	49.1	0.5	1.56	0.43	0.22	10.86
113	500	130	49.1	0.85	1.56	0.43	0.22	7.02
114	500	130	49.1	1.25	1.56	0	0	5.19
115	500	130	49.1	2	1.56	0.43	0.12	3.24
116	500	120	50.7	0.5	1.29	0.23	0.13	6.63
117	500	120	73.6	0.5	1.29	0.23	0.13	9.03
118	500	120	73.6	0.5	1.29	0.47	0.13	9.14
119	500	120	73.6	0.85	1.29	0.47	0.37	6.8
120	500	130	49.1	0.5	1.56	0.43	0.12	11.47
121	500	130	49.1	0.85	1.56	0.43	0.12	8.15
122	791	305	32	1.16	1.27	0.35	0.31	5.62
123	463	110	53.9	0.27	1.23	0	0.48	12.57
124	463	110	48.2	1.08	1.23	0	0.48	5.3
125	463	110	50.6	1.62	1.23	0	0.48	4.32
126	463	110	41.1	2.16	1.23	0	0.48	2.95
127	463	110	42.8	2.7	1.23	0	0.48	2.06
128	425	203	33.9	2.15	2.67	0	2.65	4.17
129	305	102	20.9	1	1.94	0	0	4.37
130	305	102	19	1.21	1.94	0.45	0.77	5.09
131	305	102	19.2	1.5	1.94	0.23	0.18	3.83
132	305	102	20.4	1.5	1.94	0.68	0.31	4.01
133	305	102	19.6	1.5	1.94	0.23	0.77	4.71
134	724	76	21.5	0.35	0.52	0	2.45	4.34
135	470	76	21.2	0.54	0.79	0	2.45	5.32
136	216	76	21.7	1.18	1.73	0	2.45	5.48
137	470	76	19.2	0.54	0.79	2.45	0	5.82
138	216	76	22.6	1.18	1.73	2.45	0	5.3
139	597	76	21	0.43	0.62	0.86	0	4.43
140	343	76	22	0.74	1.09	0.86	0	4.22
141	216	76	22.6	1.18	1.73	0.86	0	5.85
142	724	76	18.6	0.35	0.52	0.61	0.61	4.36
143	343	76	21.9	0.74	1.09	0.61	0.61	4.87
144	390	203	24.8	2.34	3.1	0	0.38	3.09
145	390	203	23.6	1.56	2.07	0	0.69	3.66
146	390	203	27	1.56	2.07	0	0.69	3.64
147	390	203	24.5	1.56	3.1	0	0.34	3.91
148	390	203	24.5	1.17	1.63	0	0.46	3.24
149	500	130	49.1	0.85	1.56	0.43	0.12	7.51
150	500	130	49.1	1.25	1.56	0.43	0.22	5.86
151	500	120	50.7	0.5	1.29	0.94	0.13	7.58
152	500	120	50.7	1.25	1.29	0	0.13	3.56
153	500	120	50.7	1.25	1.29	0.47	0.13	4.61

Table 7 (continued)

Case number	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
154	500	120	50.7	1.25	1.29	0.94	0.13	4.85
155	500	120	73.6	0.85	1.29	0.47	0.24	6.84
156	500	120	73.6	1.25	1.29	0	0.13	4.85
157	500	120	73.6	2	1.29	0.47	0.13	3.56
158	500	130	49.1	0.85	1.56	0.43	0.12	8.73
159	500	130	49.1	1.25	1.56	0.43	0.12	5.58
160	375	150	32.4	0.89	2.04	0.1	0.23	7.57
161	375	150	32.4	0.81	2.04	0	0	8.16
162	380	100	50.3	0.9	4.08	0.15	0.67	12.74
163	463	110	53	0.54	1.23	0	0.48	9.43
164	463	110	48	1.08	1.23	0	0.48	5.69
165	463	110	45.3	1.62	1.23	0	0.48	3.4
166	425	203	27.5	2.15	2.67	0	2.65	5.1
167	425	203	28.7	2.15	2.67	0	2.65	4.94
168	425	203	32.1	2.15	2.67	0	2.65	4.53
169	425	203	34.4	2.15	2.67	0	2.65	4.58
170	425	203	31	2.15	2.67	0	2.65	4.48
171	425	203	33.2	2.15	2.67	0	2.65	3.83

Table 8 Testing data set

Case number	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
1	305	102	18.7	1	1.94	0.23	0.28	5.18
2	305	102	19.8	1	1.94	0.45	0.63	5.48
3	305	102	18.4	1	1.94	0.23	1.25	5.18
4	305	102	17.5	1.21	1.94	0.68	0.42	4.05
5	305	102	21.8	1.5	1.94	0.91	0.18	3.93
6	305	102	19.9	1.5	1.94	0.23	0.31	3.99
7	305	102	20.8	1.5	1.94	0.91	0.31	3.99
8	305	102	21	1.5	1.94	0.23	0.56	4.53
9	305	102	19.5	2.08	1.94	0	0	2.36
10	724	76	19.2	0.35	0.52	0	0.86	4.53
11	216	76	20.1	1.18	1.73	0	0.86	6.09
12	470	76	20.1	0.54	0.79	0.61	0.61	4.84
13	390	203	23.6	2.34	3.1	0	0.38	2.64
14	390	203	23.7	1.95	3.1	0	0.37	3.6
15	390	203	23.3	1.95	3.1	0	0.37	3.39
16	390	203	26.3	1.95	3.1	0	0.73	4.07
17	390	203	26.3	1.56	2.07	0	0.34	3.93
18	390	203	24	1.56	2.07	0	0.34	3.11
19	390	203	14.1	1.56	2.07	0	0.34	2.83
20	390	203	26.2	1.17	1.63	0	0.46	3.8
21	390	203	24.5	1.17	1.63	0	0.61	4.23
22	500	130	23.7	0.85	1.56	0	0	4.08
23	500	130	49.1	0.85	1.56	0	0	6.17
24	500	130	49.1	0.85	1.56	0.43	0.34	6.47
25	500	130	49.1	2	1.56	0.43	0.34	3.62
26	500	120	50.7	0.85	1.29	0.47	0.13	6.54
27	500	120	50.7	0.85	1.29	0.47	0.24	6.01

Table 8 (continued)

Case number	X_1	X_2	X_3	X_4	X_5	X_6	X_7	Y
28	500	120	50.7	2	1.29	0.46	0.13	2.75
29	500	120	73.6	0.5	1.29	0.94	0.13	9.11
30	500	120	73.6	0.85	1.29	0.47	0.13	6.96
31	500	120	73.6	1.25	1.29	0.94	0.13	5.55
32	500	130	23.7	1.25	1.56	0.43	0.12	3.51
33	500	130	49.1	1.25	1.56	0.43	0.12	5.81
34	370	150	22	1.42	2.79	0	0	3.53
35	375	150	32.4	0.89	2.04	0.1	0.23	8.11
36	463	110	56	0.54	1.23	0	0.48	9.19
37	463	110	45.7	0.54	1.23	0	0.48	8.74
38	463	110	53.9	0.54	1.23	0	0.48	9.82
39	463	110	51.2	0.81	1.23	0	0.48	7.91
40	463	110	44	0.81	1.23	0	0.48	7.85
41	463	110	44.6	1.62	1.23	0	0.48	3.73
42	425	203	39	2.15	2.67	0	2.65	5.55
43	425	203	29.4	2.15	2.67	0	2.65	4.27

References

- Tan KH, Weng LW, Teng S (1997) A strut-and-tie model for deep beams subjected to combined top-and-bottom loading. *Struct Eng J* 75(13):215–225
- Park J-W, Kuchma D (2007) Strut-and-Tie model analysis for strength prediction of deep beams. *ACI Struct J* 104(6):657–666
- Tang C, Tan K (2004) Interactive mechanical model for shear strength of deep beams. *J Struct Eng* 130(10):1534–1544
- Pal M, Deswal S (2011) Support vector regression based shear strength modelling of deep beams. *Comput Struct* 89(13–14):1430–1439
- ACI-318 ACIC (2011) 318-11: Building Code Requirements for Structural Concrete and Commentary. American Concrete Institute
- CIRIA-Guide2 CIRaIA (1977) CIRIA Guide 2: The Design of Deep Beams in Reinforced Concrete. CIRIA, Ove Arup and Partners
- CSA CSA (1994) Design of concrete structures: structures (design)—a national standard of Canada. CAN-A23.3-94. Toronto
- Cheng M-Y, Prayogo D, Wu Y-W (2014) Novel genetic algorithm-based evolutionary support vector machine for optimizing high-performance concrete mixture. *J Comput Civil Eng* 28(4):06014003
- Cheng M-Y, Firdausi PM, Prayogo D (2014) High-performance concrete compressive strength prediction using genetic weighted pyramid operation tree (GWPOT). *Eng Appl Artif Intell* 29:104–113
- Tien Bui D, Nhu V-H, Hoang N-D (2018) Prediction of soil compression coefficient for urban housing project using novel integration machine learning approach of swarm intelligence and multi-layer perceptron neural network. *Adv Eng Inform* 38:593–604
- Hoang N-D, Bui DT (2018) Predicting earthquake-induced soil liquefaction based on a hybridization of kernel Fisher discriminant analysis and a least squares support vector machine: a multi-dataset study. *Bull Eng Geol Env* 77(1):191–204
- Shaik S, Krishna KSR, Abbas M, Ahmed M, Mavaluru D (2018) Applying several soft computing techniques for prediction of bearing capacity of driven piles. *Eng Comput*. <https://doi.org/10.1007/s00366-018-0674-7>
- Moosazadeh S, Namazi E, Aghababaei H, Marto A, Mohamad H, Hajihassani M (2019) Prediction of building damage induced by tunnelling through an optimized artificial neural network. *Eng Comput* 35:579–591
- Sharma LK, Singh TN (2018) Regression-based models for the prediction of unconfined compressive strength of artificially structured soil. *Eng Comput* 34(1):175–186
- Vapnik VN (1995) The nature of statistical learning theory. Springer, New York
- Suykens JAK, Gestel TV, Brabanter JD, Moor BD, Vandewalle J (2002) Least squares support vector machines. World Scientific Publishing Company, Singapore
- Tien Bui D, Tuan TA, Hoang N-D, Thanh NQ, Nguyen DB, Van Liem N, Pradhan B (2017) Spatial prediction of rainfall-induced landslides for the Lao Cai area (Vietnam) using a hybrid intelligent approach of least squares support vector machines inference model and artificial bee colony optimization. *Landslides* 14(2):447–458
- Chou J-S, Pham A-D (2013) Enhanced artificial intelligence for ensemble approach to predicting high performance concrete compressive strength. *Constr Build Mater* 49:554–563
- Hancock T, Put R, Coomans D, Vander Heyden Y, Everingham Y (2005) A performance comparison of modern statistical techniques for molecular descriptor selection and retention prediction in chromatographic QSRR studies. *Chemom Intell Lab Syst* 76(2):185–196
- Acar E, Rais-Rohani M (2009) Ensemble of metamodelling with optimized weight factors. *Struct Multidisc Optim* 37(3):279–294
- Wang Y-R, Yu C-Y, Chan H-H (2012) Predicting construction cost and schedule success using artificial neural networks ensemble and support vector machines classification models. *Int J Project Manage* 30(4):470–478
- Chou J-S, Lin C (2013) Predicting disputes in public-private partnership projects: classification and ensemble models. *J Comput Civil Eng* 27(1):51–60
- Chou J-S, Yang K-H, Lin J-Y (2016) Peak shear strength of discrete fiber-reinforced soils computed by machine learning and metaensemble methods. *J Comput Civil Eng* 30(6):04016036
- Prayogo D, Cheng MY, Widjaja J, Ongkowijoyo H, Prayogo H (2017) Prediction of concrete compressive strength from early age test result using an advanced metaheuristic-based machine

- learning technique. In: ISARC 2017—proceedings of the 34th international symposium on automation and robotics in construction, 2017. pp 856–863
25. Cheng M-Y, Prayogo D, Wu Y-W (2018) Prediction of permanent deformation in asphalt pavements using a novel symbiotic organisms search–least squares support vector regression. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-018-3426-0>
 26. Aljarah I, Al-Zoubi AM, Faris H, Hassonah MA, Mirjalili S, Saadeh H (2018) Simultaneous feature selection and support vector machine optimization using the grasshopper optimization algorithm. *Cognitive Comput* 10(3):478–495
 27. Faris H, Hassonah MA, Al-Zoubi AM, Mirjalili S, Aljarah I (2018) A multi-verse optimizer approach for feature selection and optimizing SVM parameters based on a robust system architecture. *Neural Comput Appl* 30(8):2355–2369
 28. Hoang N-D, Pham A-D (2016) Hybrid artificial intelligence approach based on metaheuristic and machine learning for slope stability assessment: a multinational data analysis. *Expert Syst Appl* 46:60–68
 29. Cheng M-Y, Prayogo D (2014) Symbiotic organisms search: a new metaheuristic optimization algorithm. *Comput Struct* 139:98–112
 30. Tejani GG, Savsani VJ, Patel VK (2016) Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. *J Comput Des Eng* 3(3):226–249
 31. Tejani GG, Savsani VJ, Bureerat S, Patel VK (2018) Topology and size optimization of trusses with static and dynamic bounds by modified symbiotic organisms search. *J Comput Civil Eng* 32(2):04017085
 32. Tejani GG, Savsani VJ, Patel VK, Mirjalili S (2018) Truss optimization with natural frequency bounds using improved symbiotic organisms search. *Knowl Based Syst* 143:162–178
 33. Kumar S, Tejani GG, Mirjalili S (2018) Modified symbiotic organisms search for structural optimization. *Eng Comput*. <https://doi.org/10.1007/s00366-018-0662-y>
 34. Tejani GG, Pholdee N, Bureerat S, Prayogo D (2018) Multiobjective adaptive symbiotic organisms search for truss optimization problems. *Knowl Based Syst* 161:398–414
 35. Tejani GG, Pholdee N, Bureerat S, Prayogo D, Gandomi AH (2019) Structural optimization using multi-objective modified adaptive symbiotic organisms search. *Expert Syst Appl* 125:425–441
 36. Prayogo D, Susanto YTT (2018) Optimizing the prediction accuracy of friction capacity of driven piles in cohesive soil using a novel self-tuning least squares support vector machine. *Adv Civil Eng* 2018:9
 37. Prayogo D, Cheng M-Y, Wong FT, Tjandra D, Tran D-H (2018) Optimization model for construction project resource leveling using a novel modified symbiotic organisms search. *Asian J Civil Eng* 19(5):625–638
 38. Cheng M-Y, Prayogo D Modeling the permanent deformation behavior of asphalt mixtures using a novel hybrid computational intelligence. In: ISARC 2016—33rd international symposium on automation and robotics in construction, Auburn, USA, 2016. International association for automation and robotics in construction, pp 1009–1015
 39. Cheng M-Y, Prayogo D, Wu Y-W (2018) A self-tuning least squares support vector machine for estimating the pavement rutting behavior of asphalt mixtures. *Soft Comput*. <https://doi.org/10.1007/s00500-018-3400-x>
 40. Ezugwu AE, Prayogo D (2019) Symbiotic organisms search algorithm: theory, recent advances and applications. *Expert Syst Appl* 119:184–209
 41. Cheng M-Y, Wibowo DK, Prayogo D, Roy AFV (2015) Predicting productivity loss caused by change orders using the evolutionary fuzzy support vector machine inference model. *J Civil Eng Manag* 21(7):881–892
 42. Bishop CM (2006) *Pattern recognition and machine learning (Information Science and Statistics)*. Springer, New York
 43. Chou J-S, Chiu C-K, Farfoura M, Al-Taharwa I (2011) Optimizing the prediction accuracy of concrete compressive strength based on a comparison of data-mining techniques. *J Comput Civil Eng* 25(3):242–253
 44. Hoang N-D, Tien Bui D, Liao K-W (2016) Groutability estimation of grouting processes with cement grouts using differential flower pollination optimized support vector machine. *Appl Soft Comput* 45:173–186
 45. Clark AP (1951) Diagonal tension in reinforced concrete beams. *ACI J* 48(10):145–156
 46. Kong FK, Robins PJ, Cole DF (1970) Web reinforcement effects on deep beams. *ACI J Proc* 67(12):1010–1018
 47. Smith KN, Vantsiotis AS (1982) Shear strength of deep beams. *ACI J Proc* 79(3):201–213
 48. Anderson NS, Ramirez JA (1989) Detailing of stirrup reinforcement. *ACI Struct J* 86(5):507–515
 49. Tan K-H, Kong F-K, Teng S, Guan L (1995) High-strength concrete deep beams with effective span and shear span variations. *ACI Struct J* 92(4):395–405
 50. Oh J-K, Shin S-W (2001) Shear strength of reinforced high-strength concrete deep beams. *ACI Struct J* 98(2):164–173
 51. Aguilar G, Matamoros AB, Parra-Montesinos GJ, Ramirez JA, Wight JK (2002) Experimental evaluation of design procedures for shear strength of deep reinforced concrete beams. *ACI Struct J* 99(4):539–548
 52. Quintero-Febres CG, Parra-Montesinos G, Wight JK (2006) Strength of struts in deep concrete members designed using Strut-and-Tie method. *ACI Struct J* 103(4):577–586
 53. Gandomi AH, Alavi AH, Shadmehri DM, Sahab MG (2013) An empirical model for shear capacity of RC deep beams using genetic-simulated annealing. *Arch Civil Mech Eng* 13(3):354–369
 54. Chang C-C, Lin C-J, Technology (2011) LIBSVM: a library for support vector machines. *ACM Trans Intell Syst* 2(3):27
 55. De Brabanter K, Karsmakers P, Ojeda F, Alzate C, De Brabanter J, Pelckmans K, De Moor B, Vandewalle J, Suykens JA (2010) *LS-SVMLab toolbox user's guide: version 1.7*. Katholieke Universiteit Leuven
 56. Hsu CW, Chang CC, Lin CJ (2003) *A practical guide to support vector classification*. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>