Structural and Multidisciplinary Optimization
https://doi.org/10.1007/s00158-019-02325-4

RESEARCH PAPER

A gradient boosting approach with diversity promoting measures for the ensemble of surrogates in engineering

Nestor V. Queipo [1] · Efrain Nava [1]

Received: 18 December 2018 / Revised: 28 April 2019 / Accepted: 3 June 2019

Abstract
Ensemble of surrogates has shown to be effective in the modeling of a variety of engineering problems, from aerospace to automotive and oil and gas industries, among many others. The ensemble performance though can be negatively affected by the lack of diversity (correlated errors) among the members of the ensemble, i.e., similar errors throughout the input space. During the last decade, ensemble research efforts in engineering have mostly focused on the so-called ensemble integration (combining prediction) phase with, for example, linear weighted averages of surrogate predictions based on performance (error) measures of both local and global natures. On the ensemble generation side (members of the ensemble), the emphasis has been on promoting diversity through the so-called heterogeneous ensembles, that is the use of different learners (surrogates), such as, linear regression, radial basis functions, kriging, support vector regression, and neural networks. Because of the lack of control on diversity in heterogeneous ensembles, potentially correlated error among the learners remains a critical issue. This paper presents an alternative approach based on homogeneous ensembles (single learning algorithm) that promotes diversity both implicitly and explicitly, by randomly sampling the training data and deterministically changing the data supplied to each surrogate model in the ensemble generation process. Specifically, it includes decision trees as learners, subsampling of the training set, and training on random selections of the input features, i.e., random subspace. The ensemble is sequentially constructed as an additive model where at each stage a simple surrogate is fit to the negative gradient of the loss function at the training data of the latest ensemble (gradient boosting). In addition, as a by-product of the modeling process, the proposed approach provides important measures to preliminary rank input features. Statistically significant prediction improvements over heterogeneous ensembles were observed in a variety of well-known analytical test functions and industrial case studies in the areas of petroleum engineering-net present value of oil production in a mature reservoir and structural analysis-FSAE brake pedal von Mises stress and buckling load factor.

Keywords Engineering optimization · Surrogates · Ensemble · Gradient Boosting

1 Introduction

Ensemble of surrogates (e.g., Bishop 1995; Zerpa et al. 2005; Goel et al. 2007) refers to the methods that generate several models which are combined to make a prediction in regression problems. In particular, they can be shown to be (i) statistically more accurate and robust than any single surrogate (hypothesis) and (ii) less effective when the models in the ensemble exhibit significantly correlated errors (similar errors throughout the input space). The former observation has led to applications in a variety of engineering domains and ensembles becoming a fixture among the winner approaches of data science competitions (e.g., www.kaggle.com). The latter (ii) highlights the need to promote diversity during the ensemble generation process (Bishop 1995; Brown et al. 2005a; Reeve and Brown 2018). Successful application domains include, but are not limited to, aerospace (Tenne 2013), automotive (Hamza and Saitou 2012; Fang et al. 2014), and oil and gas (Chen et al. 2007; Sanchez et al. 2008).

During the last decade, significant progress has been made in the ensemble of surrogates for engineering, in particular, in the so-called area of *ensemble integration* (combining predictions), which has focused on linear weighted averages of

Responsible Editor: Vassili Toropov

✉ Nestor V. Queipo
nqueipo@ica.luz.edu.ve; nqueipo@berkeley.edu

[1] Applied Computing Institute, Faculty of Engineering, University of Zulia, Maracaibo, Venezuela

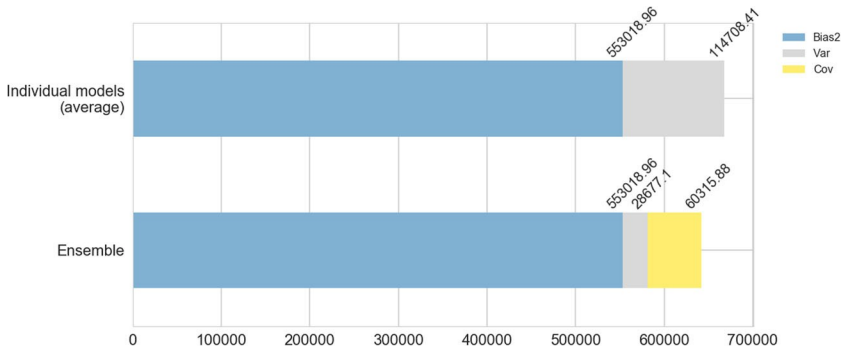


Fig. 1 Squared error bias (squared bias—Bias2), variance (Var), and covariance (Cov) decomposition for an heterogeneous ensemble of surrogates—Rosenbrock 6D test function

surrogate predictions based on performance (error) measures that are either static/global (Goel et al. 2007; Acar and Rais-Rohani 2009), dynamic/local (Zerpa et al. 2005; Sanchez et al. 2008; Zhang et al. 2012; Yin et al. 2018), or hybrid (Chen et al. 2018). In all instances, the surrogates in the ensembles were designed to have low bias (accurate) and to be diverse by adopting different learning algorithms (*heterogeneous ensembles*). For example, considering static integration, Acar and Rais-Rohani (2009) used ensembles of five different surrogates, namely, polynomial regression (PRS), radial basis functions (RBF), kriging (KR), Gaussian process (GP), and support vector regression (SVR). Audet et al. (2018), on the other hand, introduced an order-based error tailored to surrogate-based search, considering the PRS, RBF, and kernel smoothing (KS) models. In contrast, Yin et al. (2018) and Chen et al. (2018) reported ensembles with dynamic integration combining the typical metamodels, namely, PRS, RBF, and KR. Heterogeneous ensembles, tough, and lack of control of diversity with previous studies of ensembles in engineering are not explicitly testing the level of correlation among the surrogates in the *ensemble generation* phase. To avoid this shortcoming, a variety of approaches have been proposed, most notably, the so-called overproduce-and-choose with pruning process (Rooney et al. 2004; Mendes-Moreira et al. 2012) and as discussed by Bishop (1995) by accounting for the correlation among the members of the ensemble (Viana et al. 2014).

Alternatively, in homogeneous ensembles (where all the surrogates share the same learning algorithm), diversity can be more systematically controlled by randomly sampling the training data (implicit methods) or deterministically manipulating the ensemble generation process (explicit methods). Randomly sampling the data so that the surrogate models are trained with different sets of data can be very effective by, for example, sampling with replacement, i.e., bagging (Breiman 1996), and the subsampling of the training set or training on random samples of input features, i.e., random subspace (Ho 1998). On the other hand, diversity can be promoted explicitly by sequentially (Rosen 1996) or simultaneously (Liu and Yao 1999) generating models which deliberately penalize the correlation between members of the ensemble in the loss function. In addition, an ensemble can be

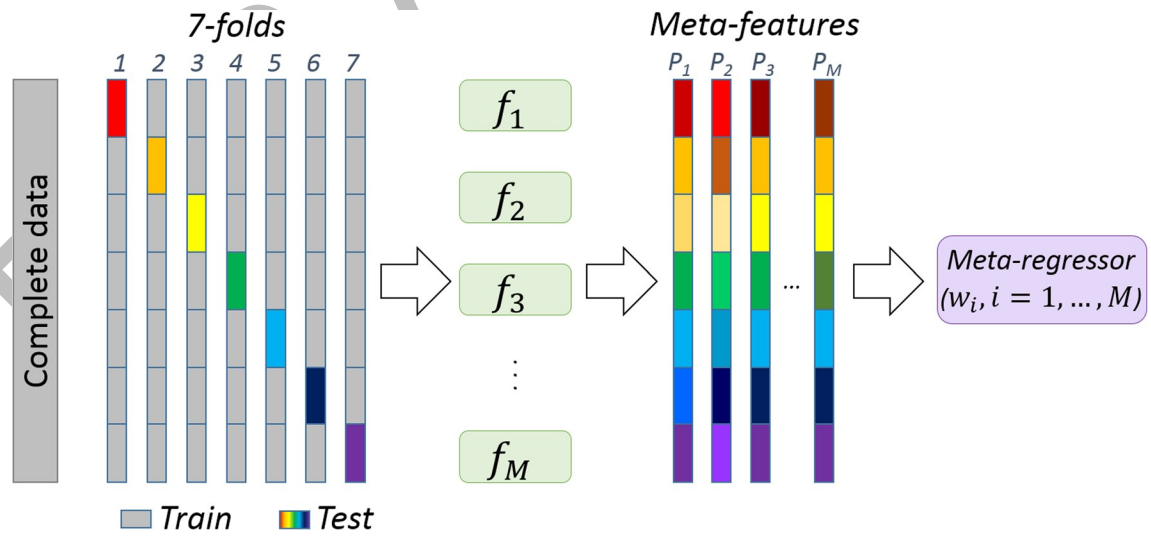


Fig. 2 Heterogeneous ensemble generation process illustrated with 7-fold cross-validation and M surrogates to construct a metalevel data (out-of-fold) set of predictions. Once the metalevel data set is available a linear regressor can be fitted; the regression coefficients represent the relative contribution of the individual surrogates to the ensemble prediction

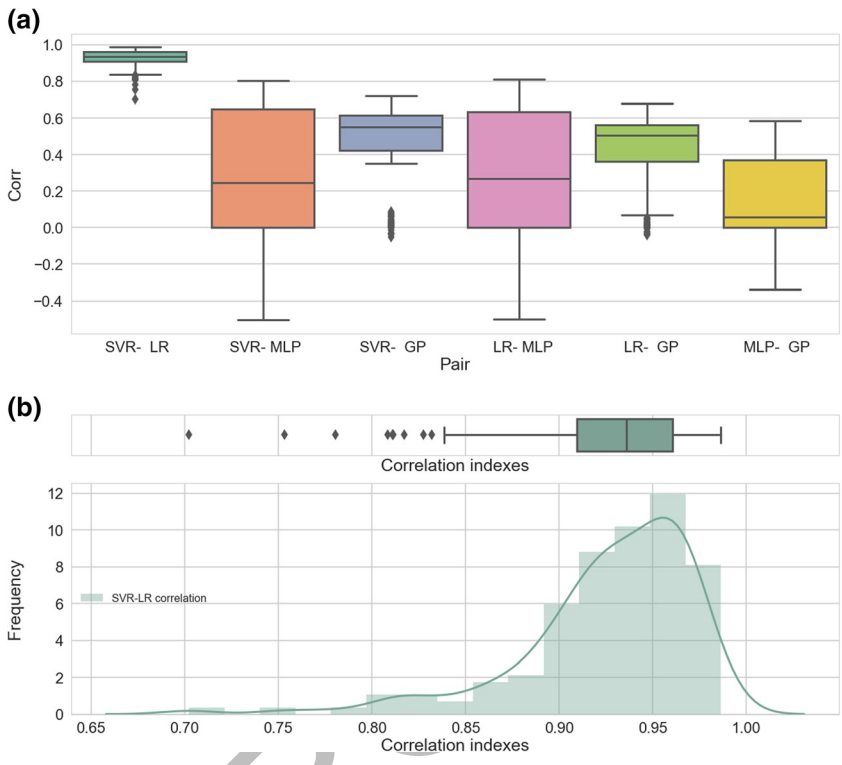


Fig. 3 Correlations among models of the ensemble (**a**) and correlations of SVR-LinearRegression, the two most correlated models (**b**)—Rosenbrock 6D test function. In this context, the labels SVR, LR, MLPR, and GP, refer to support vector, linear, multilayer perceptron, and Gaussian process regressors, respectively

constructed sequentially as an additive model where at each stage a simple surrogate (boost) is fit to the negative gradient of the loss function at the training data such that a particular cost function is minimized; for example, for a quadratic loss function, the cited gradient can be shown to be the residual of the latest ensemble. Note that the contribution (weight) of each base learner to the prediction is always dynamically established, that is, it varies with location. This latest approach is known as gradient boosting (Schapire 1990; Freund 1995), a form of functional gradient descent (Breiman 1999; Friedman 2001).

Furthermore, when using ensembles and, in general, in surrogate modeling, it is often of interest to address the so-called feature selection problem. The feature selection problem makes reference to the removal of irrelevant or redundant (nonpredictive information) inputs in the training data for faster and more effective learning. More precisely, features with higher ranking or importance scores may be chosen for further investigation or for building a more parsimonious model. A frequently adopted approach for feature selection is the Sobol method (Sobol 1993), a variance-based technique used with a trained model (as a separate task) that determines the contribution of each feature and their interactions to the overall variance of the target variable. Notably, a particular form of surrogates known as decision trees can provide importance measures to preliminary rank input variable features as a by-product of the modeling process (Karagiannopoulos et al. 2007; Kazemitabar et al. 2017).

This paper presents (i) the bias/variance/covariance formulation to raise awareness of the need to address potential correlation among surrogates in heterogeneous ensembles, (ii) an ensemble approach based on gradient boosting and decision trees with diversity promoting measures for solving engineering modeling problems, and (iii) the relative performance of the proposed approach with respect to heterogeneous ensembles on selected case studies. The remainder of this paper is structured as follows: problem statement (Section 2), bias/variance/covariance formulation (Section 3), heterogeneous ensembles and correlation among surrogates (Section 4), decision trees for regression (Section 5), a gradient boosting approach for the ensemble of surrogates (Section 6), case studies (Section 7), results and discussion (Section 8), and conclusions and recommendations (Section 9).

2 Problem statement

The problem of interest is one of typical regression, that is, given a training set D consisting of n instances, such as $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where x_i is an instance of a vector of features and y_i is a target; the task is to learn an approximate function $\hat{f} : \mathrm{X} -> \mathrm{R}$ of the true function f from D. The approximate function $\hat{f}$ can be a single surrogate f_i or the result of a linearly weighted sum of M individual surrogates F. The

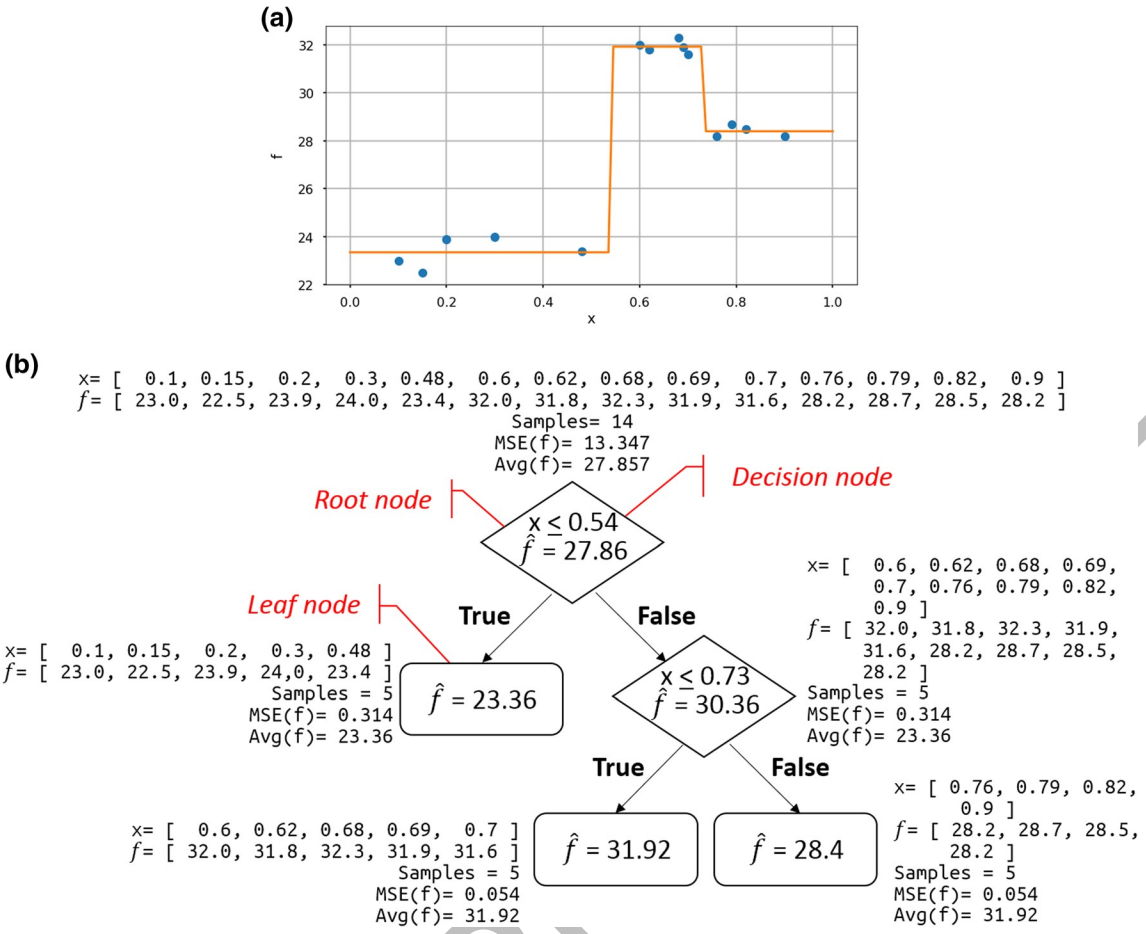


Fig. 4 A decision tree estimation from training sample (**a**) and the corresponding decision tree is also shown (**b**). Note root, decision, and leaf nodes (predictions)

function $\hat{f}$ is estimated by optimizing a loss function typically mean square error considering out-of-fold predictions.

3 Bias, variance, and covariance decomposition

For an ensemble F (with constant weights) it can be shown (Ueda and Nakano 1996) that the expected value of the MSE-mean squared error (generalization error) can be expressed as Eq. (1):

$$E\left\{(F-y)^2\right\} = \overline{\text{bias}^2} + \frac{1}{M}\,\overline{\text{var}} + \left(1-\frac{1}{M}\right)\overline{\text{covar}} \tag{1}$$

where

$$\overline{\text{bias}} = \frac{1}{M}\sum_i (E\{f_i\}-y)^2 \tag{2}$$

$$\overline{\text{var}} = \frac{1}{M}\sum_i E\left\{(f_i-E\{f_i\})^2\right\} \tag{3}$$

$$\overline{\text{covar}} = \frac{1}{M(M-1)}\sum_i\sum_{j\neq i} E\left\{(f_i-E\{f_i\})\left(f_j-E\left\{f_j\right\}\right)\right\} \tag{4}$$

where, var denotes the variance of the individual surrogates in the ensemble, and covar the averaged covariance of the surrogates in the ensemble; the estimates of all the components are computed considering several random trials (training sets). Note that in this context, diversity between surrogates (differences in the predictor outputs) is measured by their pairwise covariance. This decomposition tells us that the generalization error depends not only on the bias and variance of the individual surrogates but also on the covariance between the surrogates. The sought ensemble (optimum generalization error) should account for all components, deliberately promoting diversity among surrogates while minimizing bias.

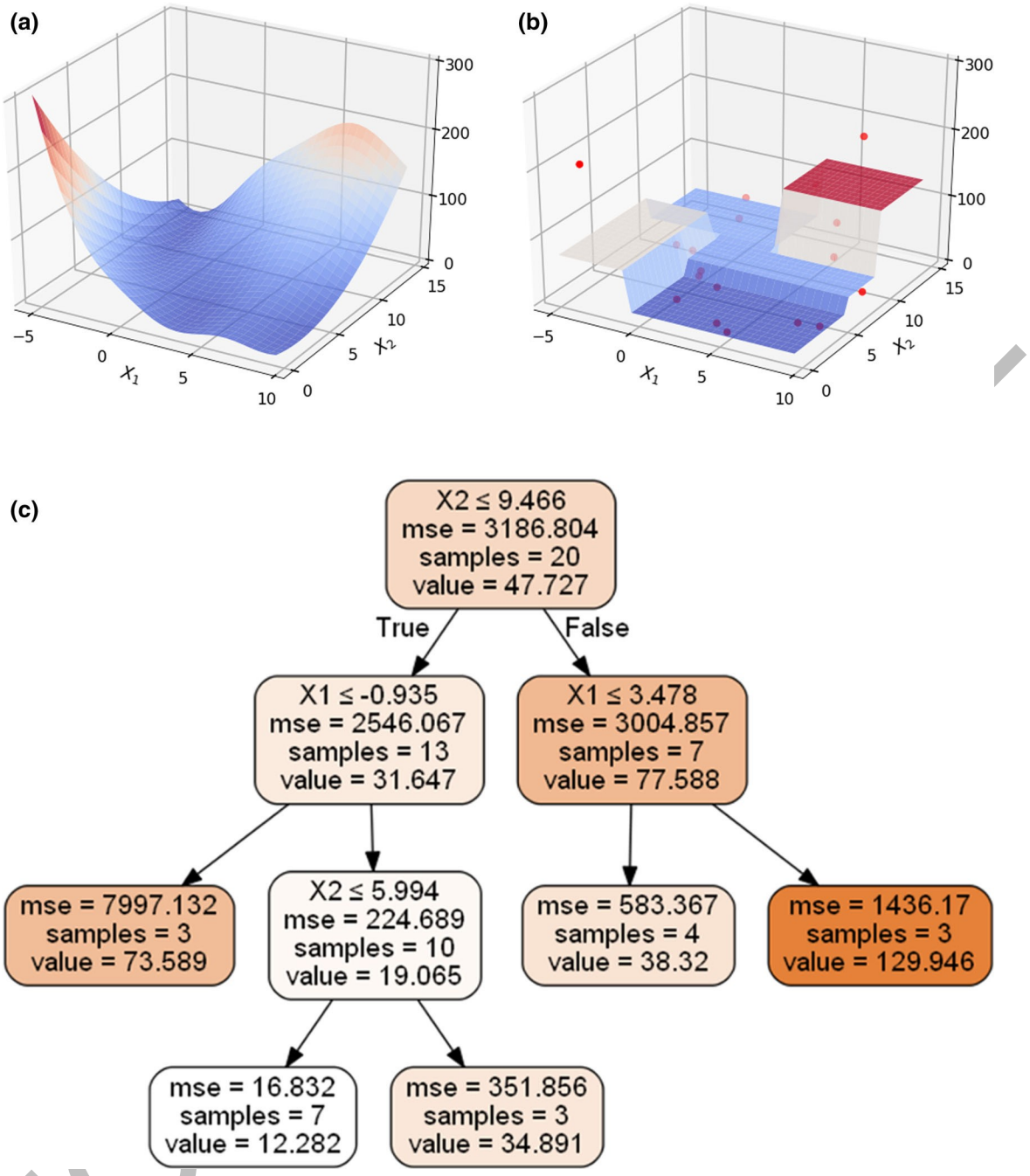


Fig. 5 Branin and Hoo 2D modeling using a decision tree. **a** Original test function. **b** Decision tree-weak learner approximation. **c** Corresponding decision tree-weak learner

Figure 1 reports the generalization error and bias, variance, and covariance decomposition for an heterogeneous ensemble of surrogates designed to approximate the well-known Rosenbrock 6D test function (Section 6), considering sixty (60) training data points and an one hundred and fifty (150) random trials. Details of the ensemble approximation are available in the next section (Section 4). As expected, the ensemble provides a lower generalization error than the average of the individual surrogates through a reduction of the error variance. Most notably, the often ignored covariance component is shown as a prominent contributor to the generalization error, only second to the bias error.

Algorithm – Simple boosting algorithm for regression

Inputs: Training set $= \{(x_1, y_1), \ldots, (x_n, y_n)\}$
M = number of sub-models (base learners) of the ensemble

Output: F_M = the ensemble

1: let h_0 be a constant model
2: let F_0 be an ensemble just consisting of h_0
3: **for** $m = 1, \ldots, M$
4: **for** each pair (x_i, y_i) in the training set
5: compute the residual R (negative gradient) $R(y_i, F_{m-1}(x_i))$
6: **end for**
7: train a regression sub-model h_m on the residuals
8: add h_m to the ensemble and find the optimal parameter α for each of leaf of h_m; $F_m = F_{m-1}(x) + \alpha_m \cdot h_m(x)$
9: **end for**
10: **return** the ensemble F_M

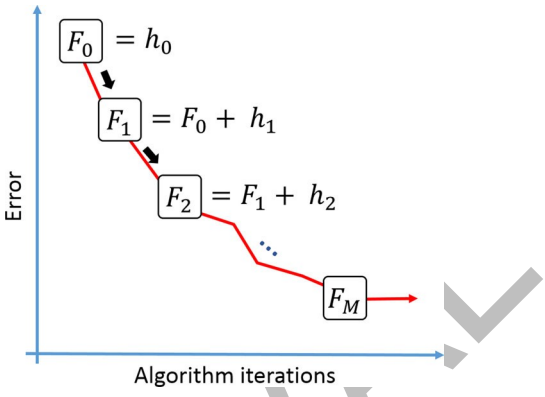


Fig. 6 Basic boosting algorithm for building ensembles assuming a quadratic loss function. The initial constant model F_0 can be, for example, the mean of the training target values

4 Heterogeneous ensembles and correlation between surrogates

These ensembles combine a set of heterogeneous surrogates, i.e., generated by different learning algorithms L_i on a single data set D; each of the surrogates has their hyperparameters that must be chosen among a given set of candidates (hyperparameter grid), for example, the number of neurons in the hidden layer in a multilayer perceptron neural network (MLP). In this study, we focus on ensembles based on linear weighted averages of surrogate predictions considering static/global performance measures (e.g., MSE, R^2). In abstract terms, their basic construction strategy can be described using the following steps:

1. Specify the set of possible hyperparameters considering all the members of the ensemble.
2. Split data set D into k folds, with training sets D^{-i} corresponding to $D-$ fold i, and $i = 1, 2, \ldots, k$.
3. For each of the elements in the set of hyperparameters:
   a. Generate a set of M base-level learners (surrogates) $f_1, \ldots, f_M$ using (inner) cross-validation.
   b. Make out-of-fold predictions with the M trained base-level learners (surrogates).
   c. Train (identify the weights) a metalevel linear regressor using the outputs (predictions) from step 3.b and the corresponding target values.

Table 1 Modeling techniques (including the proposed approach) under consideration and the corresponding computer implementations (libraries)

Model	Implementation
Ensembles	
GBwDM (proposed approach)	Scikit-learn http://scikit-learn.org/stable/
Heterogeneous ensemble approach (HEA)	MLxtend https://rasbt.github.io/mlxtend/
Base learners	
Linear regression	Scikit-learn http://scikit-learn.org/stable/
SVR-radial basis function	
Multilayer perceptron	
Gaussian process	

Table 2 Hyperparameters in the ensembles throughout the case studies

Ensemble	Hyperparameter	Values
GBwDM	Learning rate	0.05, 0.10
	Number of estimators	50, 75, 100
	Subsampling	0.3, 0.5, 1.0
	Feature sampling	$\sqrt{\#\text{features}}$, #features
	Tree's max depth	2, 3, 4
HEA	Base learners	Linear regressor
		SVR RBF
		ML perceptron
		Gaussian process

4. Select the metalevel linear regressor among those obtained in 3.c corresponding to the lowest (outer) cross-validation error computed on D^i.

Figure 2 illustrates the training process for the ensemble using (inner) cross-validation. The features of the training set for the metaregressor are P_1, P_2,.., P_M,that is, the out-of-fold predictions (P_i) of the base-level learners (f_i). The Appendix. Metamodels presents a brief description of each of the surrogates under consideration, a combination of the parametric and nonparametric models, namely, linear regression (13.1), multilayer perceptron neural networks (13.2), support vector regression-RBF (13.3), and Gaussian processes (13.4).

In general, though, there has been no explicit evaluation of the correlation among the individual surrogates which as previously discussed may compromise the generalization errors of heterogeneous ensembles. An illustration of how members of the ensembles can in fact be correlated follows.

The correlation between surrogates f_i and f_j can be estimated through the expression in Eq. (5) as follows:

$$\text{corr}\left(\boldsymbol{x}, y, f_i, f_j\right) = \frac{E\left\{\left(y - f_i\right)\left(y - f_j\right)\right\}}{\sigma\left(f_i\right)\sigma\left(f_j\right)} \tag{5}$$

Using the heterogeneous ensemble cited in the previous section (Rosenbrock 6D test function), Fig. 3 a displays the empirical distribution (box plots) of correlation values for different pairs of surrogates in one hundred and fifty (150) random trials. Note that the SVR-Linear regression pair exhibits considerably high median correlation values (over 0.8). Details of the empirical distribution observed between Support Vector Machine - Linear regression are available in Fig. 3b. It further illustrates how different learning algorithms in heterogeneous ensembles do not necessary lead to the desired outcome—negative or noncorrelated models.

5 Decision trees for regression

A decision tree learner, a nonparametric supervised method, builds regression models in the form of a tree structure (Breiman et al. 1984). The goal: predict the value of a target variable by learning simple decision rules inferred from the data features. Starting from the *root* (with the whole training data), data sets are broken down (split) at *decision nodes* into smaller and smaller subsets considering at each split the feature with maximal reduction of a weighted (with the number of samples) variability measure (e.g., mean square error).

Table 3 Formulas for the selected analytical test functions with number of features and range

Function	No. of dimensions	Formula	Range
Branin and Hoo Dixon and Szegö 1978	2D	$\left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10$ (7)	$5 \le x_1 \le 10$ $0 \le x_2 \le 15$
Rosenbrock 6D Rosenbrock 1960	6D	$\sum_{i=1}^{N/2}\left[100\left(x_{2i-1}^2 - x_{2i}\right)^2 + \left(x_{2i} - 1\right)^2\right]$ (8)	$-2 \le x_i \le 2$
Dixon and Price 12D Dixon and Price 1989	12D	$(x_1 - 1)^2 + \sum_{i=2}^{12}\left(2x_i^2 - x_{i-1}\right)^2$ (9)	$-10 \le x_i \le 10$ $i = 1, 2, \ldots, 12$

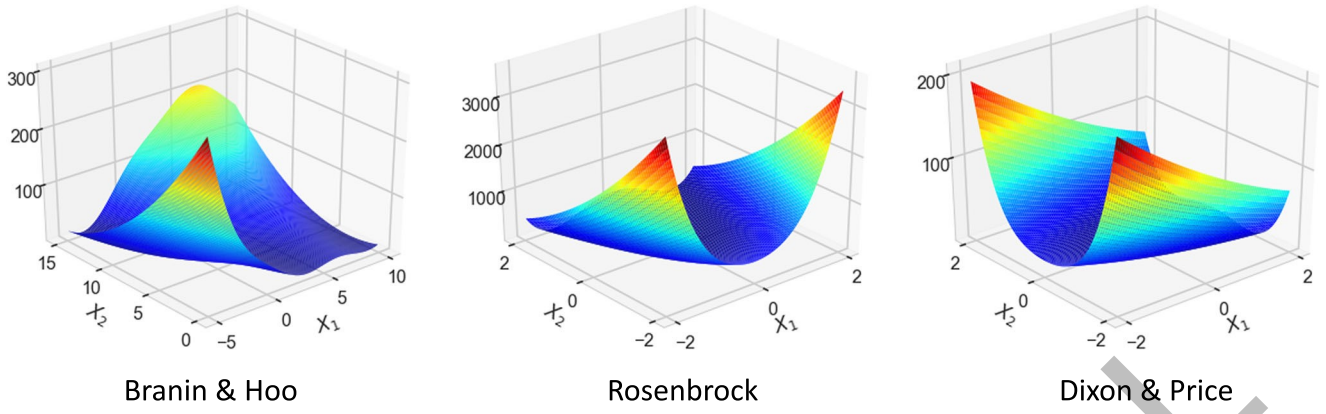


Fig. 7 Two-feature representations of the selected analytical test functions

After a termination criterion is met, e.g., a prespecified maximum depth is reached, the result is a tree with decision and leaf nodes. A decision node has two (binary) branches, each associated with a set of values for the tested feature, while a leaf node represents a prediction (average value of the samples at the node) of the numerical target, i.e., function output. Figure 4 illustrates a decision tree with root, decision, and leaf nodes for a single variable (feature) function; note that at each decision node, a splitting led to a maximum reduction in the weighted standard deviation of the data.

6 Algorithm

A decision tree for regression can be obtained through the following steps (with mean square error adopted as a variability measure):

I. The whole training data set is considered as the root; compute the mean square error of the outputs in the training data set.
II. The data set is split considering the different attributes into child nodes. At this step, for each of the features under consideration, the data set is split, and the mean square error for the data in the child nodes is calculated and weighted with the number of samples in each node. The resulting mean square error—weighted sum of MSE among the two branches—is subtracted from the one before the split, and hence, the MSE reduction is obtained. The standard candidate split points are those halfway between consecutive pairs of the sorted input data; see Fayyad and Irani (1992) for a more efficient approach.
III. Feature selection for decision nodes. Considering the features with the largest MSE reduction are

Table 4 Base learner hyperparameter grid—analytical test functions

Model	Hyperparameter	Values		
		Branin and Hoo	Rosenbrock	Dixon and Price
Linear regression	–	–	–	–
SVR-radial basis function	C*	55, 110, 220	900, 1800, 3600	382, 764, 1528 ($\times 10^3$)
	Gamma	0.01, 0.10, 0.20	0.100, 0.125, 0.150	0.1, 0.2, 0.5 ($\times 10^{-3}$)
Multilayer perceptron	Hidden layer size**	3, 5, 6	4, 6, 8	4, 6, 9
	Activation function	ReLU, logistic	ReLU, logistic	ReLU, logistic
Gaussian process	–	–	–	–

*The C and ϵ values under consideration are 0.5, 1.0 and 2.0 times the reference values calculated as proposed by Cherkassky and Ma (2004)

**Settled by limiting the number of parameters of the model (weights) to be a fraction (40%, 60%, 80%) of the total number of data points available during the learning process

Table 5 NPV formulae (target function) with economic parameters and the discrete range of potential well locations

Net Present Value [Bs.]

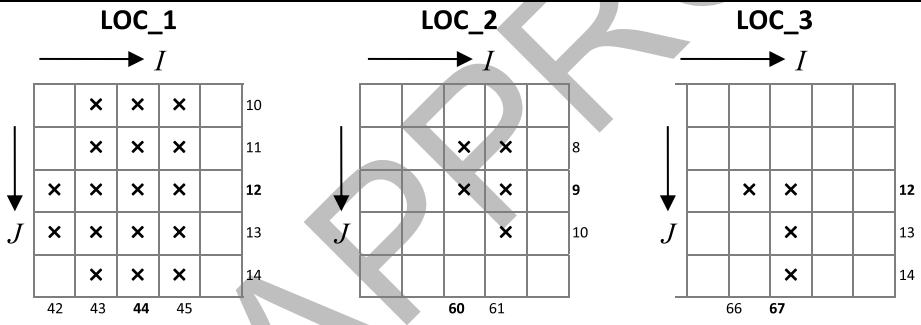
$$\sum_{k=1}^{K} \frac{\left(q_o(k)P_o + q_g(k)P_g - q_w(k)C_w - q_i(k)C_i - OC_k\right)\Delta t_k}{(1+\beta)^{\frac{k\Delta t_k}{365}}} \quad (10)$$

where

		Value	Unit
$q_{\{o,g,w\}}$	daily production of oil, gas and water	-	[bbl]
q_i	water injection rate	-	[bbl]
P_o	net income due to oil production	See Table 6	[Bs/bbl]
P_g	net income due to gas production	4 030.65	[Bs/MMBTU]
C_w	cost of handling the water production	0	[Bs/bbl]
C_i	water injection cost	40.10	[Bs/bbl]
OC_k	Operating costs(e.g., drilling) at the *k*th instant	-	[Bs]
k	sampling instant	-	[days]
Δt_k	is the time step size in days	-	[days]
β	annual discount factor	10	[%]
K	production horizon (20 years) in days	20*365	[days]

Bbl: barrels; Bs.: Bolivars, the Venezuelan currency

Range of potential well locations



associated with decision nodes, a measure of the importance of a particular feature can then be calculated as the sum of the MSE reductions over all nodes where a split was made on that feature; note that the MSE reductions are weighted to account for the size (number of samples) of the node. Important features can appear at different depths of the trees since a particular splitting variable may or not be good in terms of overall accuracy.

IV. Recursively process (II–III) nonleaf nodes until a termination criterion is met. Each branch in the previous step leads to another decision node (further splitting will be conducted) or a leaf node (e.g., MSE below a threshold value). In practice, a termination criterion is needed, for example, when the MSE for a branch is a small fraction (e.g., 5%) of the MSE associated with the training data (root node) or the number of samples in the leaf nodes reach a minimum value (e.g., 3). When there is more

Table 6 Oil price profile used in the net present value calculation

Year	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027
Price	41.63	51.56	56.39	60.19	64.00	65.74	68.54	69.49	71.49	73.49	74.49	75.49
Year	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	
Price	76.49	77.49	78.49	79.49	80.49	81.49	82.49	83.49	84.49	85.49	86.49	

Table 7 General mature reservoir characteristics—NPV of mature oil reservoir case study

Property	Value
Initial pressure, psia	5550
Bubble pressure (Pb), psia	4215
Average porosity, fraction	0.18
Average permeability, mD	294.9
Average rock shale (Vsh), fraction	0.25
Initial water saturation (Swi), fraction	0.28
API gravity	39.5
Oil volumetric factor (Boi), bbl/STB	1.7668
Initial solution gas-oil ratio (Rsi), SCF/STB	1436
Oil viscosity at Pb, cps	0.251

than one sample in a leaf node, the prediction for the output is the average of the sample values.

Decision tress have several advantages; for example, it allows for both discrete and continuous variables, naturally couples model training with preliminary feature importance, and represents a white box model easy to understand and interpret. Unfortunately, they can create overcomplex models that do not generalize the data well (overfitting) that are based on a greedy algorithm where locally optimal decisions are made at each node which can result in far from optimal trees, and small variations in the data might result in very different outcomes. Figure 5 shows the Branin and Hoo 2D original test function (a), the decision tree-weak learner approximation (b), and the corresponding decision tree-weak learner (c) obtained using a training sample size of twenty (20) and a maximum depth of three (*weak* learner). The shortcoming of decision trees are addressed by sequentially training multiple (weak) trees in an ensemble learner; details of such an approach are next.

7 A gradient boosting approach for the ensemble of surrogates

The proposed approach, labeled as GBwDM (gradient boosting with diversity measures), is designed to promote diversity among base learners, hence reducing the covariance component of the generalization error and potentially improving prediction estimates, be robust to overfitting, and, as a by-product of the modeling process, generate preliminary importance measures of the feature variables. We claim these objectives can be achieved by adopting a homogeneous ensemble approach based on decision trees and gradient boosting considering subsampling and random subspace strategies. The architecture (components and interactions) of the approach is described below.

I. Hyperparameters setting. In this stage, a set of hyperparameters for the ensemble of surrogates is defined. In the case of ensembles of decision trees using gradient boosting, hyperparameters are, for example, number of trees, maximum depth in trees, subsample and random subspace values, number of estimators, and learning rate. In the spirit of implicitly promoting

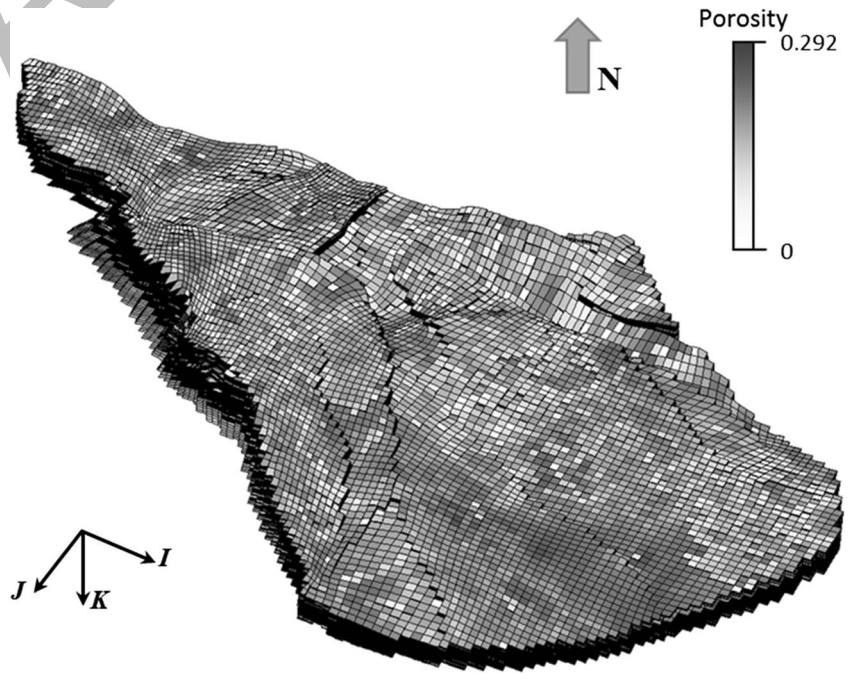


Fig. 8 Numerical grid of mature reservoir simulation model—oil production NPV case study

Table 8 Hyperparameter grid for the heterogeneous ensemble—oil production NPV case study

Model	Hyperparameter	Values
Linear regression	–	–
SVR-radial basis function	C*	135×10^6, 270×10^6, 540×10^6
	Gamma	0.10, 0.15, 0.20
Multilayer perceptron	Hidden layer size**	4, 6, 8
	Activation function	ReLU, logistic
Gaussian process	–	–

*The C and ϵ values under consideration are 0.5, 1.0 and 2.0 times the reference values calculated as proposed by Cherkassky and Ma (2004)

**Settled by limiting the number of parameters of the model (weights) to be a fraction (40%, 60%, 80%) of the total number of data points available to the learning process

diversity, subsampling makes reference to fitting the base learners with a fraction (subsample value) of the training data, and the random subspace value specifies the number of features to consider when looking for the best split. The hyperparameters are selected from a grid of prespecified values (grid search) using *k*-fold cross-validation.

II. Ensemble generation. Figure 6 shows the basic steps of using the gradient boosting algorithm with a quadratic loss function. Note it builds the ensemble by sequentially fitting a base learner (each function increment is called a boost), e.g., a *weak* decision tree, to the residuals of the latest ensemble ($y_i - F_{m-1}(x_i)$), that is, the negative gradient of the loss function at the training data (explicitly promoting diversity). *Weak* decision trees, also called *stumps*, refer to the models where a rough approximation is sought and often translates to limiting the trees depth to a very low number (often 2 to 3).

While in this study the base learners are decision trees and the loss function is quadratic, gradient boosting is not limited to a particular surrogate or loss function. More generally, at each iteration, the base learners h_m are trained on the so-called pseudo-residuals, which represent the negative gradient Eq. (6) of the loss function at training points.

$$\nabla L = \frac{\partial L(y_i, F_{m-1}(x_i))}{\partial F_{m-1}(x_i)} \tag{6}$$

That is, the loss reduction for each training point if the predicted value—$F_{m-1}(x_i)$—was to become one unit closer to the target value. While the base learner loses accuracy at the training points, it provides an approximation of the cited gradient throughout the input space—$h_m(x)$—which is used to implement a greedy stagewise strategy for calculating the latest ensemble—$F_m(x)$—i.e., $F_m(x) = F_{m-1}(x) + \alpha_m \cdot h_m(x)$. Finally, at each leaf, the optimal parameter α is sought so that $L(y_i, F_m(x_i))$ is minimized.

While the number of base learners are a parameter to be set, the gradient boosting ensembles have been shown to be resistant to overfitting so it can be specified quite robustly without jeopardizing predictive performance. An efficient and scalable implementation of the gradient boosting framework known as extreme gradient boosting, aka XGBoost, is also available (Chen and Guestrin 2016).

III. Ensemble selection. Considering the set of hyperparameters in (I), the best ensemble is the one with the lowest score in holdout or nested cross-validation (Cawley and Talbot 2010). When holdout (data split intro training and test sets) is not used, nested cross-validation should be implemented so that the same data is not used to both tune the ensemble hyperparameters and evaluate the ensemble performance, leading to overfitting and underestimation of the prediction errors. Nested CV effectively uses a series of train/validation/test set splits, where in the inner loop, ensembles are fitted to training data, and the best set of hyperparameters is the one associated with the ensemble with the lowest score over the validation set. In the outer loop, the best ensemble is identified as the one with the lowest (average) estimated generalization error over test sets. The particular approach to ensemble selection should be identified depending on the data size, model stability, and among other factors.

IV. Feature importance. Feature importance refers to the reduction in model accuracy when a variable is removed, and it is critical for variable selection and the building of the parsimonious and interpretable models. As previously mentioned, decision trees can naturally couple the training process with preliminary feature importance identification at no additional computational expense. In this study, the overall importance score for a particular feature is the average of the ones obtained in the base learners (stumps) in the ensemble.

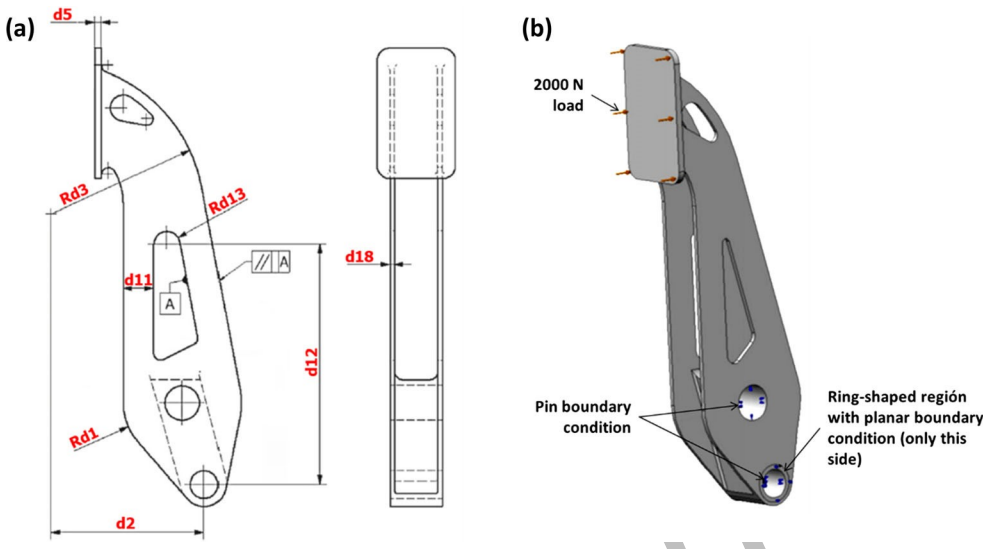


Fig. 9 Geometric variables (**a**) and loads and boundary conditions (**b**)—FSAE brake pedal case study

8 Case studies

The proposed approach is evaluated using the three analytical test functions, namely, Branin and Hoo 2D (Dixon and Szegö 1978), Rosenbrock 6D (Rosenbrock 1960), and Dixon and Price 12D (Dixon and Price 1989) and two industrial case studies in the areas of engineering modeling, i.e., the von Mises stress and buckling load factor in a Formula SAE[1] brake pedal (structure analysis) as discussed in Romero and Queipo (2017) and the net present value of the 20-year oil production of a mature oil reservoir considering alternative new well locations (reservoir engineering).

8.1 General considerations

Table 1 lists the ensembles and individual surrogate models (base learners) under consideration and the corresponding computer implementations (libraries). The ensembles are identified as gradient boosting with diversity measures (GBwDM-proposed approach) and heterogeneous ensemble (HEA); the individual members of the heterogeneous ensemble are also indicated, i.e., linear regression, SVR-radial basis function, neural networks, and Gaussian process. All the results were obtained using the cited libraries within Python programs in Google's Colaboratory environment—a free Jupyter notebook environment than runs entirely in the Cloud. Jupyter is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text (www.jupyter.org).

[1] Formula SAE is a student design competition organized by SAE International, previously known as the Society of Automotive Engineers, SAE.

All the models were trained/tested using the holdout cross-validation approach; the general hyperparameters (Table 2) of all the models were identified using 5-fold cross-validation within the training set, and their values were the result of a preliminary study and adopting common practices. For all case studies, the size of the training sets was specified as 10 times the number of dimensions, and seeking sample-independent results, one hundred and fifty (150) random trials of the training/test sets were conducted.

The results of the proposed approach and those of heterogeneous ensembles are compared considering median of the mean square error in the test data set; the number of times each particular type of ensemble exhibited a lower mean square error (winner) is also reported.

8.2 Analytical test functions

The selection of these functions is aimed at providing a variety of function behavior, number of features, and ranges; Branin and Hoo (2D), Rosenbrock (6D), and Dixon and Price (12D) all frequently used as modeling and optimization case studies (Table 3). Figure 7 illustrates the two-feature behavior of the analytical test functions under consideration.

8.2.1 Training and test data sets

For all the analytical case studies, data is generated using a Latin hypercube design for both training and test sets, with sizes 10× and 100× the corresponding input domain dimension, respectively. For example, for the Rosenbrock 6D case study, the training and test sets are of sizes 60 and 600, respectively. On a separate note, Table 4 shows the hyperparameter grid under consideration for the heterogeneous ensemble.

Table 9 Feature probability distributions—FSAE brake pedal case study

Feature	Distribution	Parameters
$d1$	Uniform	[27, 33]
$d2$		[70, 80]
$d3$		[70, 87]
$d5$		[2, 4]
$d11$		[8, 18]
$d12$		[80, 115]
$d13$		[2, 8]
$d18$		[2, 6]
E	Lognormal	μ (11.059)
		σ (0.0709)

8.3 Industrial test cases

These cases differ in the nature of the simulation-based models, i.e., multiphase flow in porous media, and structural analysis and the number and type of features under consideration, that is, 6D discrete and 9D continuous input variables, respectively.

8.3.1 Net present value of the oil production in a mature reservoir considering alternative new well locations

Most of the current oil production comes from mature reservoirs, that is, reservoirs whose production have reached their peaks and have started to decline. At the same time, the rate of replacement of the produced reserves by new discoveries of conventional resources has been steadily declining in the last decade, with a trend that will continue for the foreseeable future (Syed 2012, Brown et al. 2017). Hence, the modeling and optimization of oil recovery processes from aging resources are rapidly becoming a pressing issue for both energy policy and oil producers.

The problem of interest is to estimate the net present value (NPV) of the oil production in a mature reservoir (in western Venezuela) during a 20-year horizon considering the variables LOC_1i, LOC_1j, LOC_2i, LOC_2j, LOC_3i, and LOC_3j, that is, the potential areal location of three (3) new wells. The NPV (target variable) formulae and the discrete range of potential well locations are available in Table 5; note the NPV is a function of the reservoir production (oil + gas), water injection, and operating costs under a specific oil price profile (Table 6). The case study corresponds to a real reservoir whose main characteristics are summarized in Table 7.

The reservoil numerical model has a grid of $135 \times 75 \times 78$ for a total of 789,785 cells, with an average cell size of 100 m × 100 m × 3 m (Fig. 8). This model corresponds to a mature reservoir with more than 80 wells and a 20-year exploitation plan that includes among other elements three (3) new wells to be drilled at specific times. The wells are assumed to be vertical, and the NPV of interest is associated with alternative aerial locations for the new wells.

8.3.2 Training and test data sets

The data consists of 178 samples randomly divided in training and test data sets of size 60 and 118, respectively. Table 8 shows the hyperparameter grid for the heterogeneous ensemble.

8.4 FSAE brake pedal

The Formula SAE project (SAE international 2016a, b, c) with the annual participation of 2470 students from 120 university teams promotes careers and excellence in engineering in the automotive industry including

Table 10 Hyperparameter grid for the heterogeneous ensemble—FSAE brake pedal case study

Base learner	Hyperparameter	Values	
		von Mises stress	Buckling load factor
Linear regression	–	–	–
SVR-radial basis function	C*	60, 120, 240	10, 20, 40
	Gamma	0.10, 0.15, 0.20	0.10, 0.15, 0.20
Multilayer perceptron	Hidden layer size**	4, 6, 9	4, 6, 9
	Activation function	ReLU, logistic	ReLU, logistic
Gaussian process	–	–	–

*The C and ϵ values under consideration are 0.5, 1.0 and 2.0 times the reference values calculated as proposed by Cherkassky and Ma (2004)

**Settled by limiting the number of parameters of the model (weights) to be a fraction (40%, 60%, 80%) of the total number of data points available to the learning process

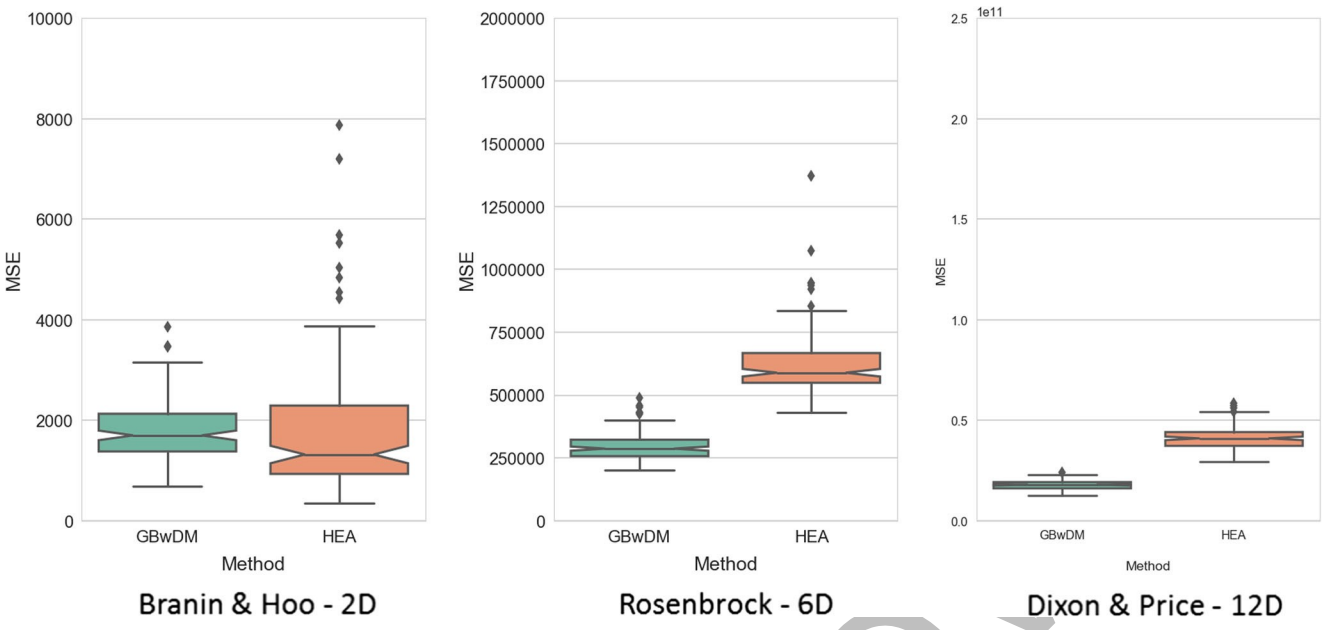


Fig. 10 Test set MSE distributions for the analytical case studies, i.e., Branin and Hoo—2D, Rosenbrock—6D, and Dixon and Price—12D

research, design, manufacturing, testing, developing, marketing, management, and finances. In particular, each component in a single-seater FSAE vehicle must be safe and lightweight in order to decrease the vehicle mass and take the most advantage of the engine power. The brake pedal design of FSAE vehicles represents an easy-to-understand structural problem among engineering students and practitioners around the world yet relevant to aerospace and automotive industrial environments.

With reference to Fig. 9a, the problem of interest is to estimate the von Mises stress (ε_S) and buckling load factor (ε_B) in a FSAE brake pedal considering as features eight (8) geometric variables (in circles) and the Young modulus E of the material. The loads and boundary conditions of the brake pedal are also shown in Fig. 9b with the range of the features specified in Table 9. Please see Romero and Queipo (2017) for a detailed discussion.

8.5 Training and test data sets

The data consists of 1320 samples randomly divided in training and test data sets of sizes 90 and 1230, respectively.

Table 10 shows the hyperparameter grid for the heterogeneous ensemble.

9 Results and discussion

The proposed approach (GBwDM) exhibited the following:

- The lowest median (statistically significant) of test set mean square errors and most winners for analytical (except the 2D) and some industrial case studies, specifically, oil production NPV optimization and FSAE – von Mises stress
- Strong resistance to overfitting for all case studies

Table 11 Median of test set MSE for the analytical case studies, i.e., Branin and Hoo—2D, Rosenbrock—6D, and Dixon and Price—12D

MSE (test set)	Branin and Hoo 2D	Rosenbrock 6D	Dixon and Price 12D
GBwDM	1538.59	283, 234.06	19.77×10^9
HEA	681.21	366, 008.06	20.96×10^9
Pearson's chi-squared test (p value)	$<< 0.05$	$<< 0.05$	$<< 0.05$

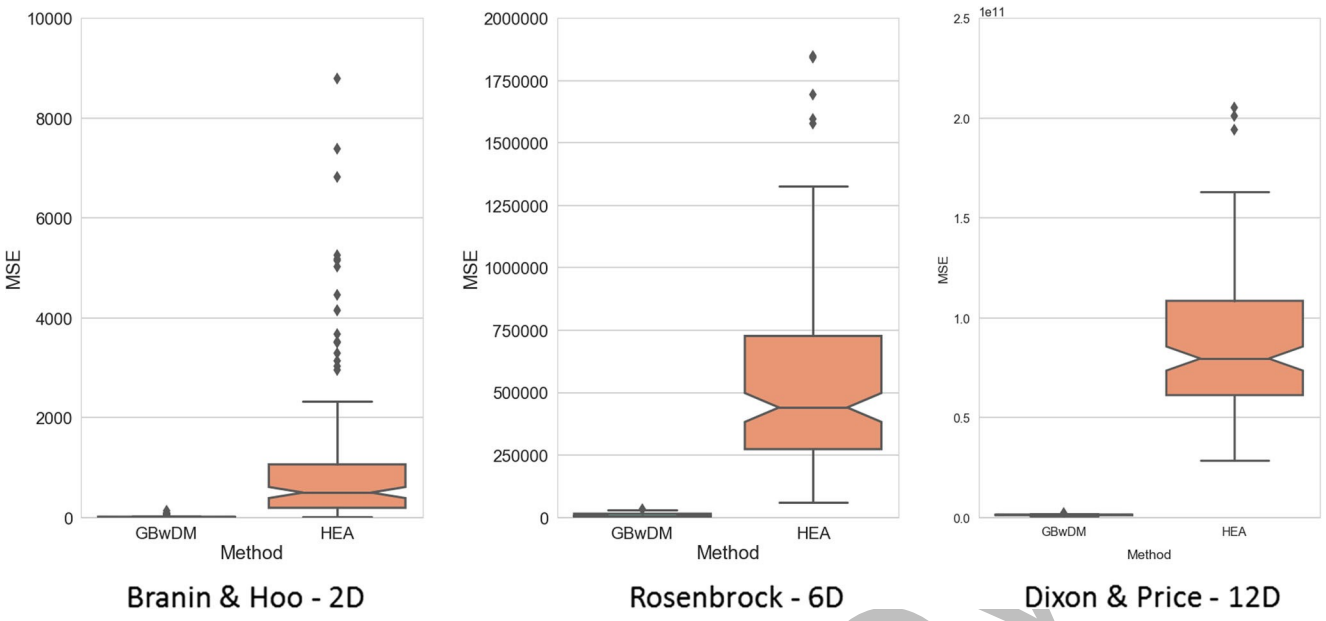


Fig. 11 Training MSE distributions for the analytical case studies, i.e., Branin and Hoo—2D, Rosenbrock—6D, and Dixon and Price—12D

- Considerable speedup with respect to heterogeneous ensemble learning
- Diversity measures that proved to be effective for improving the prediction performance
- A preliminary assessment of feature importance that is statistically aligned with those reported by the frequently used Sobol method

The lowest median of test set mean square errors for all analytical (except the 2D) case studies. Based on one hundred and fifty (150) random trials, GBwDM outperformed the heterogeneous ensemble approach (HEA) with 56% (6D) and 71% (12D) lower median MSE (Fig. 10; Table 11); the error distribution associated with the training data is shown in Fig. 11. Pearson's chi-squared test on the difference of the

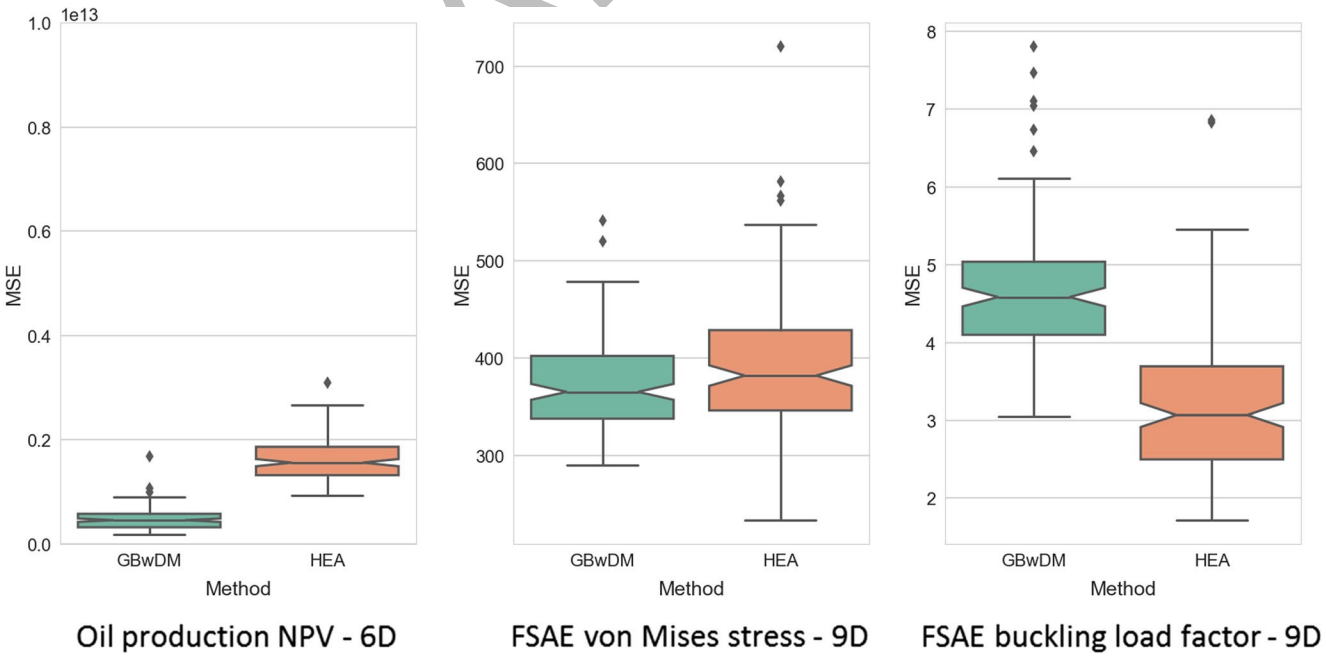


Fig. 12 Test set MSE distribution for the industrial case studies, i.e., oil production NPV, FSAE-von Mises stress, and FSAE-buckling load factor

Table 12 Median of test set MSE for the industrial case studies, i.e., oil production NPV, FSAE-von Mises stress, and FSAE-buckling load factor

MSE (test set)	Oil production NPV (5D)	FSAE-von Mises stress (9D)	FSAE-buckling load factor (9D)
GBwDM	0.827×10^{11}	389.34	4.58
HEA	1.638×10^{11}	479.25	6.75
Pearson's chi-squared test (p value)	<< 0.05	<< 0.05	<< 0.05

medians confirmed the statistical significance of the results. When considering the most meaningful analytical test cases (6D and 12D) and each of the trials, GBwDM was the absolute winner (exhibited the lowest mean square error in all instances).

As expected, the proposed approach outperformed the average error of the individual surrogates. For example, in the highest dimensional analytical case study (Dixon 12D), in the training set, the median of the MSE of the proposed approach is 51× lower than the average of the medians of the individual surrogates (consistent with the ambiguity principle, see Brown et al. 2005b) trained using 5-fold cross-validation. It is worth noting that the individual surrogate with the best performance (GPR) can be shown to overfit the data (highest test error). The second best individual surrogate (SVR) does perform only slightly better than the rest of the surrogates, but no criterion is available to select which individual surrogate will generalize the best based on training performance. Finally, MLP and LR exhibited similar performance, significantly worse than GPR. On the other hand, when considering the test set the median of the MSE of the proposed approach is 4× lower than the average of the medians of the individual surrogates. Among the individual surrogates, the best performance was provided by RBF, followed by MLP and LR with GPR exhibiting the worst performance.

GBwDM also exhibited the lowest median MSE in the oil production NPV and FSAE – von Mises stress industrial case studies. Its application resulted in median MSE 70.9% and 4% lower than those associated with the HEA (Fig. 12; Table 12) for these two case studies. Figure 13 shows the mean square errors obtained during the training phase for the industrial test cases. As with the analytical case studies, Pearson's chi-squared test on the difference of the medians confirmed the statistical significance of the cited test results. Furthermore, GBwDM was the winner (exhibited the lowest mean square error) in proportions to 148:2 (70:1) and 84:66 (9:7), respectively.

As expected, in the HEA, no single surrogate prevailed throughout all the case studies with support vector regression (SVR) and linear regression (LR) being the best performers

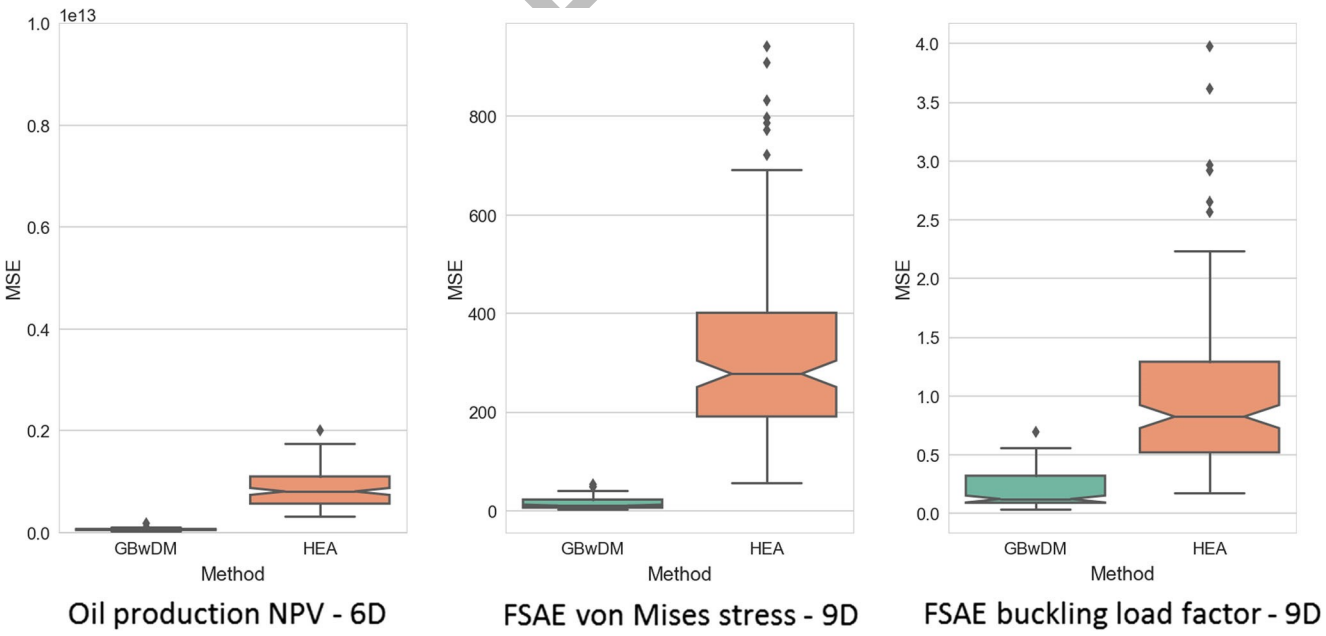


Fig. 13 Training MSE distribution for the industrial case studies, i.e., oil production NPV, FSAE-von Mises stress, and FSAE-buckling load factor

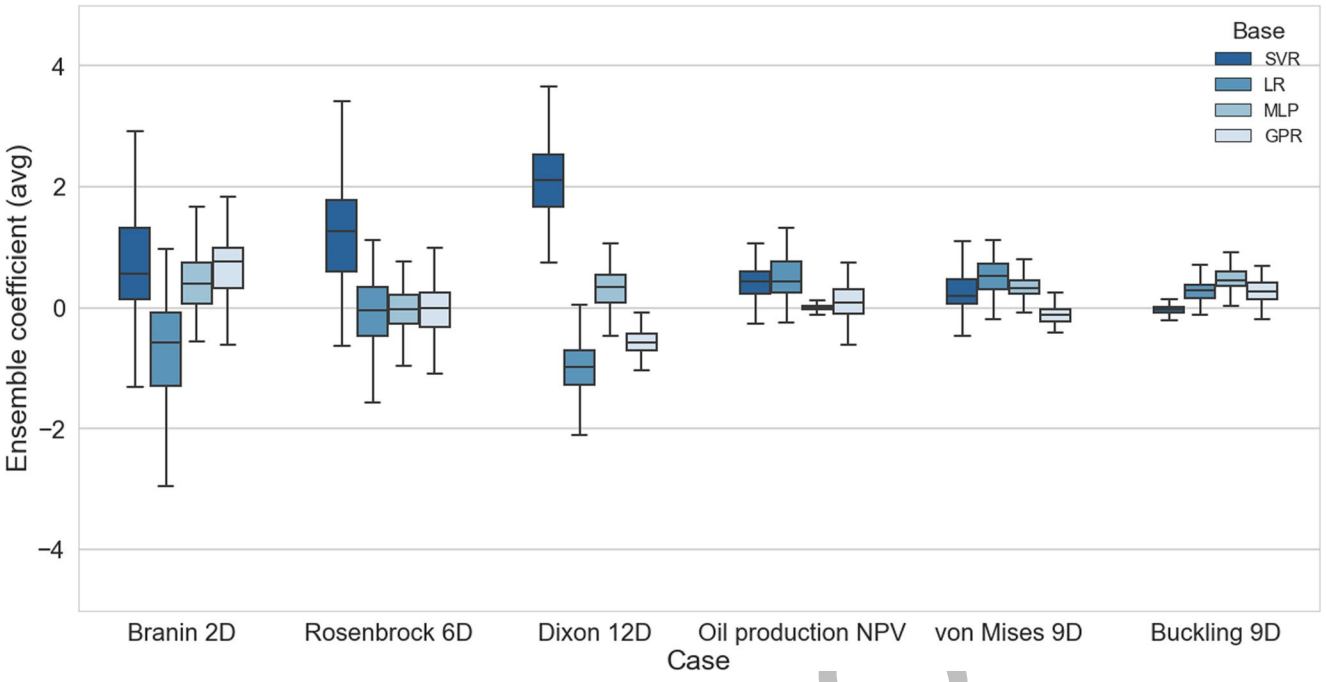


Fig. 14 Base learner coefficient (weights) distribution for the analytical and industrial case studies

and Gaussian process (GP) and multilayer perceptron (MLP) exhibiting significantly less contributions to the ensemble outputs. Specifically, SVR was the best performer in all the analytical case studies, while LR outperformed the rest of the base learners in the Oil production NPV and FSAE – von Mises stress industrial case studies (Fig. 14). Note that the weights (coefficients) in the ensemble can be negative and correspond to the implementation of heterogeneous ensembles in one of the most recognized machine learning libraries (scikit-learn.org).

GBwDM resistance to overfitting. Increasing the numbers of estimators in GBwDM invariably reduced the training error and did not lead to a significant increase in the test set error; this adds to the empirical evidence that gradient boosting can be quite robust to overfitting. This is an attractive feature considering that you can grossly specify the number of members of the ensemble and still deliver a strong prediction performance. Figures 15 and 16 depict the monotonically decreasing behavior of the training and test set error versus the number of estimators in the ensemble for all the analytical and industrial case studies, respectively.

Effectivity of diversity measures. GBwDM incorporates two promoting diversity measures in the ensemble, namely, random subspace and subsampling. Results show that in the 6D and 12D analytical case studies, a significant amount of subsampling (0.6) was adopted by most of the winners in each of the trials, approximately 62% for each case study. Similarly, random subspace (square root of the number of features) was present in a considerable amount of the winners, i.e., 64% and 47%, respectively. On the other hand, in the industrial test cases, the most effective diversity measure was subsampling. Subsampling values of 0.6 were present in 65%, 73% and 77% of the predictor winners in the oil production NPV, FSAE-von Mises stress, and FSAE-buckling load factor, respectively.

Computational efficiency. The median computational execution times of the proposed approach, in general, were significantly lower than those of the heterogeneous counterpart. For example, when considering the highest dimensional analytical test case (Dixon 12D), the computational time of the proposed approach was approx. 4 seconds versus 56 seconds of the heterogeneous ensemble approach, that is, 14× faster when using the cloud computing platform Colaboratory by Google. The main reason for the speedup can be attributed to the fact that being resistant to overfitting is a natural feature of GBwDM (stagewise parameter setting, see Hastie et al. 2009), while in the HEA, the quest for such a feature relies on a costly nested cross-validation procedure (Cawley and Talbot 2010) as implemented in MLxtend (https://rasbt.github.io/mlxtend/).

Preliminary feature selection as a by-product of the modeling process. The potential of GBwDM for this task is illustrated with the FSAE brake pedal case studies. Figure 17 a and Fig. 18 a display the ranking of the feature variables for a

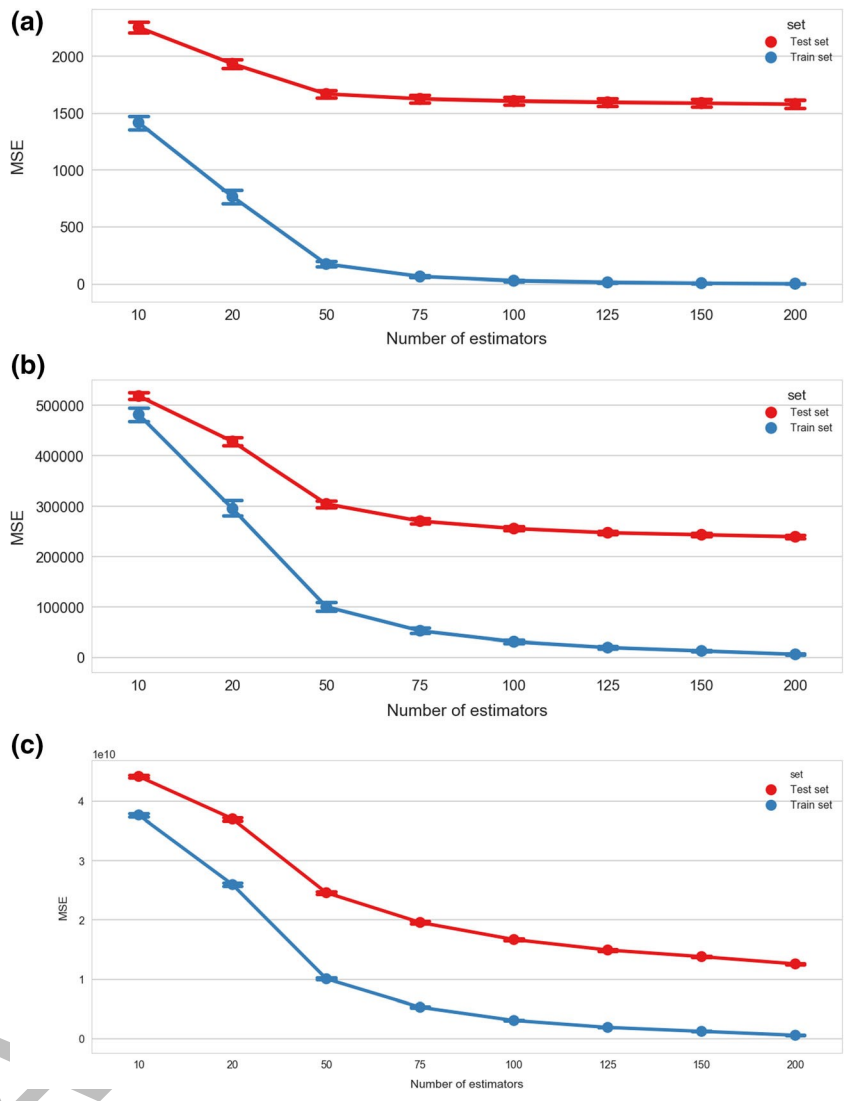


Fig. 15 Median of training and test set MSE for the analytical case studies, i.e., Branin and Hoo—2D (**a**), Rosenbrock—6D (**b**), and Dixon and Price—12D (**c**)

random trial of the training set in the FSAE-von Mises stress and FSAE-bucking load factor studies, respectively. Note that for the former target variable, the three (3) most important features (with reference to Fig. 17a) are $d18$ (69%), $d1$ (12.1%), and $d12$ (5.2%), while for the latter (Fig. 18a), those are $d18$ (95.2%), the module of elasticity E (2.1%), and $d12$ (1.4%); in parenthesis, the reduction percentage of the total MSE. Considering the learning algorithm for decision trees greedily select the splitting variables for maximum reduction of a variability measure (e.g., MSE), one would assume that the feature ranking should be related to that of the Sobol method. That indeed seems to be the case. For one hundred and fifty (150) random trials of the training set, Fig. 17 b and Fig. 18 b show the empirical distribution of a measure of the rank correlation-similarity (Kendall's tau statistic) of the ordering of the most important features obtained from GBwDM and the Sobol method for both target variables of interest. The median of the corresponding p values confirms the statistical dependence of the feature rankings from GBwDM and the Sobol method (Fig. 17c; Fig. 18c). Specifically, for the target variables of interest, the medians of the p values are in both cases approximately 1×10^{-3}, significantly lower than those of the threshold of 0.05 for rejecting the null hypothesis (the two ranking features are independent).

10 Conclusions and recommendations

This work presented a homogeneous ensemble approach based on gradient boosting and decision trees with diversity measures (subsampling and random subspace) for solving engineering modeling problems. The proposed approach is designed to improve prediction estimates by promoting diversity among the members of the ensemble (reducing the covariance component of the generalization error), be robust to overfitting, and, as a

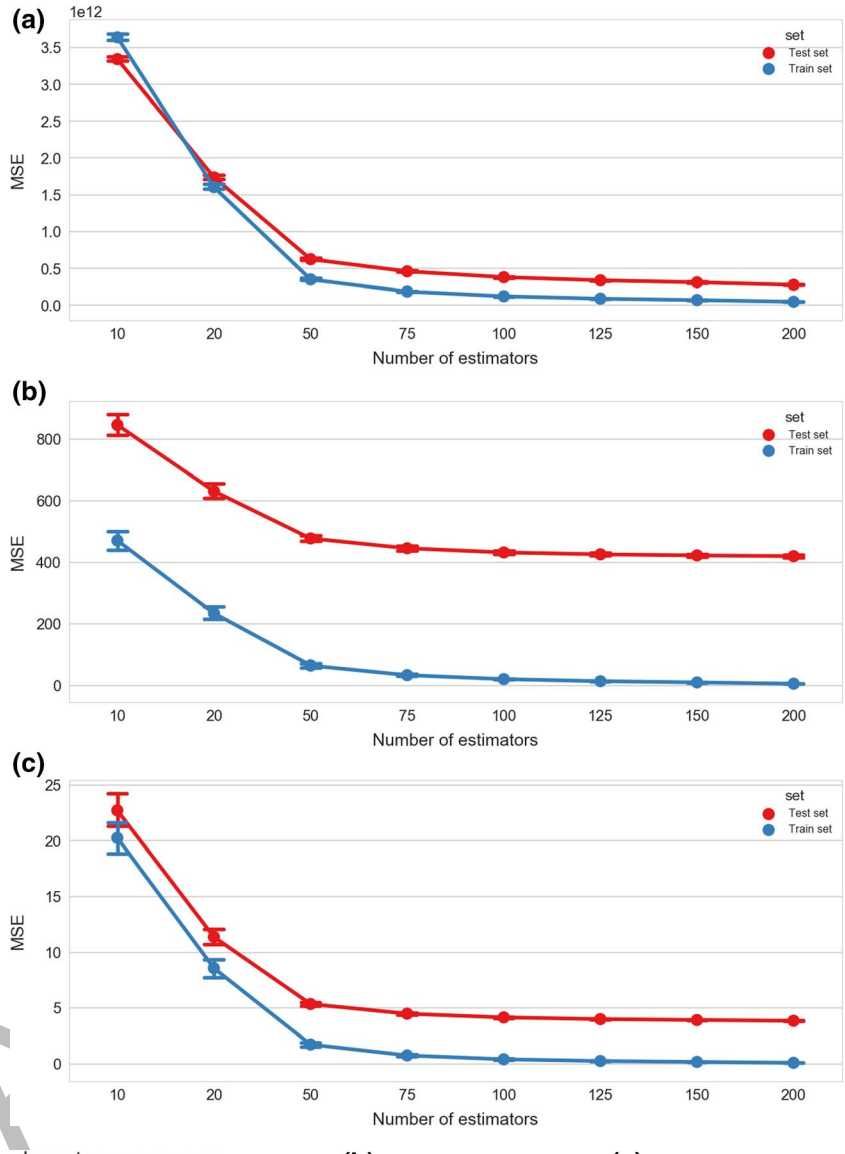


Fig. 16 Median of training and test set MSE for the industrial case studies, i.e., oil production NPV—6D (**a**), FSAE-von Mises stress—9D (**b**), and FSAE-buckling load factor—9D (**c**)

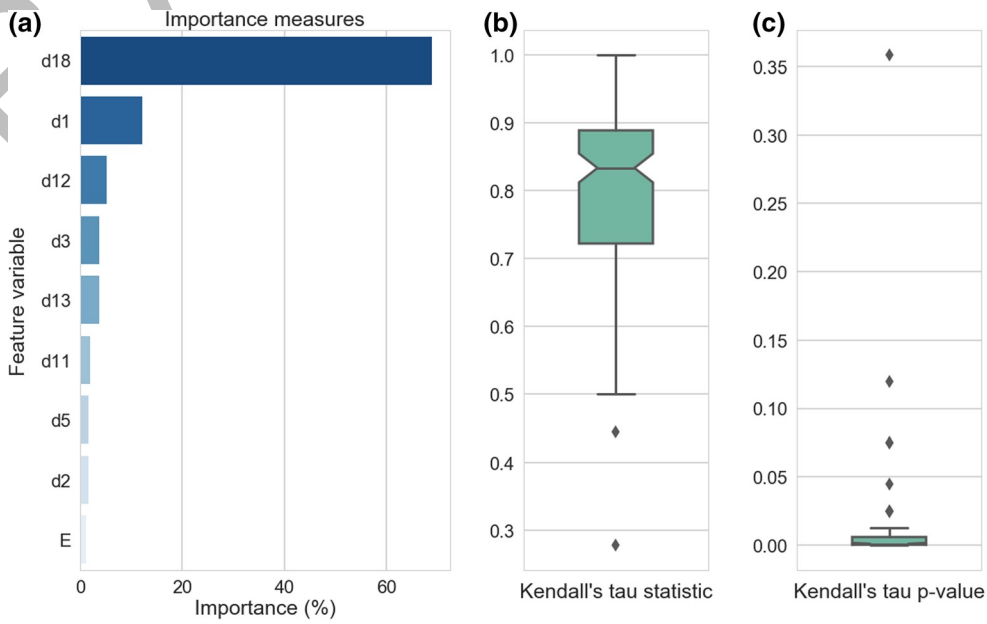


Fig. 17 GBwDM feature importance for a selected random trial (**a**), a rank correlation-similarity (Kendall's tau) statistic distribution (**b**), and the corresponding *p* value (**c**), for one hundred and fifty (150) random trials when compared with Sobol's feature importance ranking—FSAE-von Mises stress case study

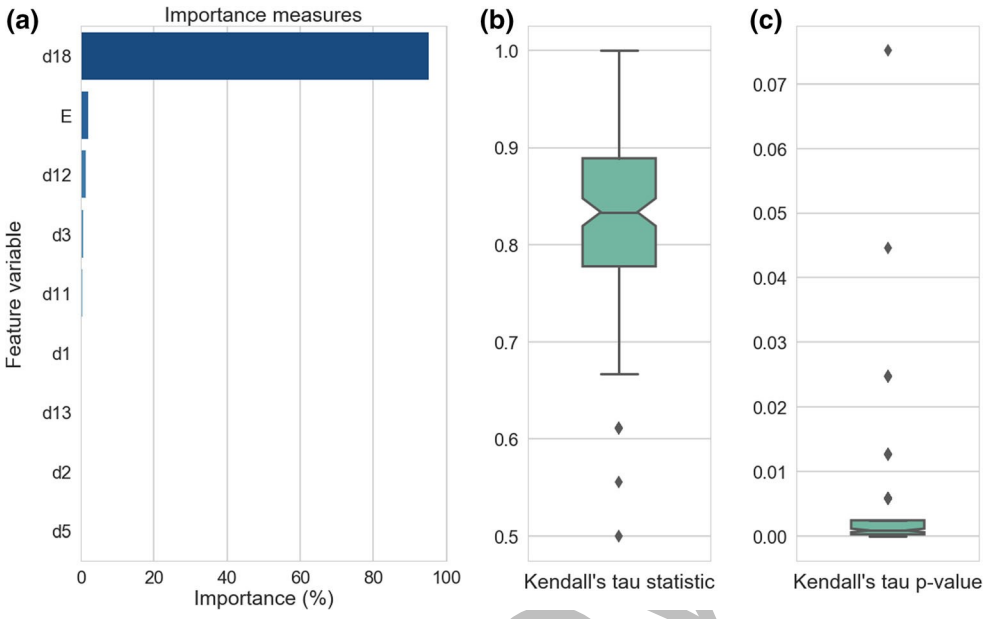


Fig. 18 GBwDM feature importance for a selected random trial (**a**), a rank correlation-similarity (Kendall's tau) statistic distribution (**b**), and the corresponding p value (**c**), for one hundred and fifty (150) random trials when compared with Sobol's feature importance ranking—FSAE-buckling load factor case study

by-product of the modeling process, generate preliminary importance measures of the feature variables.

The effectiveness of the proposed approach was demonstrated in the modeling of a variety of analytical and industrial case studies with statistically significant gains over the classical heterogeneous ensemble approach and holds promise to be particularly useful in scenarios where surrogate selection and overfitting may be critical issues. Specifically, GBwDM exhibited the lowest median of test set mean square errors and most winners for all analytical (except the 2D) and industrial case studies (except the FSAE buckling load factor). It also showed strong resistance to overfitting and diversity measures that proved to be effective for improving the prediction performance. In addition, it offered an assessment of feature importance that is statistically aligned with those reported by the frequently used Sobol method and a significant speedup over heterogeneous ensembles.

The feature importance provided by the proposed approach is only a preliminary assessment and should not be interpreted as a substitute of the application of the Sobol method, but as a by-product of the ensemble learning process. While both are the result of a variance-based global sensitivity analysis, Sobol's method represents a more theoretical sound alternative. The latter can explicitly account for the contributions to variance not only of individual variables (like in the proposed approach) but also of their interactions. Note that if a detailed global sensitivity analysis is considered necessary, in the proposed approach, the Sobol method could be coupled with the final ensemble.

This work was designed with the perspective of more systematically controlling the diversity among the members of the ensemble. The strategy: randomly sampling the training data (random subspace and subsampling) and manipulating the ensemble generation process (gradient boosting). Considering the adopted perspective and strategy (independent of surrogates and cost function) and the promising results, the proposed approach and meaningful variants should be further evaluated. In particular, some potential research venues include its application to a broader set of industrial case studies, the use of alternative weak learners, cost functions, and diversity mechanisms (e.g., penalizing in the loss function the correlation among members of the ensemble).

11 Replication of results

Data and code (Jupyter notebooks) needed to replicate the results of selected case studies can be found at https://github.com/nqueipo/SMO-Manuscript-GBwDM. The data and code are released under Creative Commons CC BY-SA and GNU General Public (GPLv3) licenses, respectively.

Table 13 Kernel functions associated with a variety of modeling schemes

Kernel	Parametrization
Polynomial order d	$k(\boldsymbol{x},\boldsymbol{x}') = (\langle \boldsymbol{x},\boldsymbol{x}' \rangle + c)^d \quad d \in N, \quad c \geq 0$
Spline	$k(\boldsymbol{x},\boldsymbol{x}') = 1 + \langle \boldsymbol{x},\boldsymbol{x}' \rangle + \frac{1}{2}\langle \boldsymbol{x},\boldsymbol{x}' \rangle \min(\boldsymbol{x},\boldsymbol{x}') - \frac{1}{6}\min(\boldsymbol{x},\boldsymbol{x}')$
RBF	$k(\boldsymbol{x},\boldsymbol{x}') = \exp(-\lVert \boldsymbol{x}-\boldsymbol{x}' \rVert^2/2h^2) \quad h > 0$

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Appendix. Metamodels

13.1. Linear regression

In this work, the regression models (Draper and Smith 1969) considered are of linear form:

$$\hat{y}(\boldsymbol{x}) = \beta_0 + \sum_{i=1}^{n_{rv}} \beta_i x_i$$

where n_{rv} is the total number of features, x_i, and the parameters β_0 and β_i are estimated by the least squares method. The implementation used in this study corresponds to *LinearRegressor* in the Scikit-learn library.

13.2. Artificial neural networks

The neural network model used corresponds to the so-called multilayer perceptron whose output can be expressed as follows:

$$O_P(I_P) = G[W \times (G[V \times I_P])]$$

where I_P denotes the input values, and V and W are the weight matrices between the input and hidden layers, and the hidden and output layers, respectively. The symbol G represents the application of the logistic function or the rectified linear unit (ReLU) to each of the elements of its arguments.

It can be shown that the three-layer perceptron is, for function approximation purposes, as capable as another neural network with higher number of layers and can estimate a function with arbitrary precision provided the number of neurons in the hidden layers and the weights are properly set (Hecht-Nielsen 1989). As a result, our attention will be restricted to the three-layer perceptron.

The number of neurons in the input and output layers is established by the number of input and output variables of the function to be estimated. The number of neurons in the hidden layer can be settled through different criteria, for example, limiting the number of parameters of the model (weights) to be a fraction of the total number of data points available to the learning process. The implementation used in this study corresponds to *MLPRegressor* in the Scikit-learn library.

13.3. Support vector regression

The kernel-based regression models $M_i s$ can be seen as solutions of the following variational problem:

$$\min_{M \in H} Z(M) = \frac{1}{n} \sum_{i=1}^{n} L(y_i - M(x_i)) + \lambda \|M\|_H^2$$

over some large space of functions H where L and λ denote a particular loss function (e.g., quadratic, Laplace, ϵ-insensitive, and Huber loss functions) and a regularization parameter, respectively. The operator $\|\bullet\|_H^2$ is the Hilbert space norm which penalizes models that are too complex. If we restrict ourselves to reproducing kernel Hilbert spaces (RKHS), the variational problem can be formulated as stated as follows:

$$\min_{M \in H} Z(M) = \frac{1}{n} \sum_{i=1}^{n} L(y_i - \langle M, K_{x_i} \rangle) + \lambda \langle M, M \rangle_H$$

It can be shown that independently of the form of the loss function, the solution of the variational problem can be expressed as follows:

$$M(x) = \sum_{i=1}^{n} \alpha_i k(x, x_i)$$

where k represents a kernel function. Table 13 shows kernel functions associated with a variety of surrogate modeling schemes. In particular, if the loss function is quadratic, the coefficients α_i can be found by solving the following linear system:

$$(n\lambda \boldsymbol{I} + \boldsymbol{K})\alpha_i = y_i$$

where $\boldsymbol{K}$ denotes the so-called Gram matrix with component K_{ij} denoting $k(x_i, x_j)$ and $\boldsymbol{I}$ representing the identity matrix. Alternatively, if the ϵ-insensitive loss function is used, the coefficients α_i are found by solving a quadratic programming problem. See Schölkopf and Smola (2002) and Poggio and Smale (2003) for details.

The implementation used in this study corresponds to *SVR* in the Scikit-learn library.

13.4 Gaussian processes

More specifically, a kriging model (Sacks et al. 1989) postulates a combination of a polynomial model $f(\boldsymbol{x})$ and an error model of the form $Z(\boldsymbol{x})$ as follows:

$$\hat{y}(\boldsymbol{x}) = f(\boldsymbol{x}) + Z(\boldsymbol{x})$$

where $\hat{y}(\boldsymbol{x})$ is the unknown function of interest, $f(\boldsymbol{x})$ is a regression function of $\boldsymbol{x}$, and $Z(\boldsymbol{x})$ is a random function (stochastic process) with mean zero and nonzero covariance given by the following expression:

$$\mathrm{Cov}\left[Z(x_i, x_j)\right] = \sigma^2 \boldsymbol{R}(x_i, x_j)$$

where σ^2 is the process variance and $\boldsymbol{R}$ is the correlation function. In this work, consideration is given to constant

regression and Gaussian correlation functions. The parameters of the regression and correlation functions are identified using a maximum likelihood estimator. The implementation used in this study corresponds to *GaussianProcessRegressor* in the Scikit-learn library.

References

Acar E, Rais-Rohani M (2009) Ensemble of metamodels with optimized weight factors. Struct Multidiscipl Optim 37(3):279–294

Audet C, Kokkolaras M, Le Digabel S, Talgorn B (2018) Order-based error for managing ensembles of surrogates in mesh adaptive direct search. J Glob Optim 70(3):645–675. https://doi.org/10.1007/s10898-017-0574-1

Bishop C (1995) Neural networks for pattern recognition. Oxford University Press, Oxford

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984). Classification and regression trees. The Wadsworth and Brooks-Cole statistics-probability series, Monterey

Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140. https://doi.org/10.1023/A:1018054314350

Breiman L (1999) Prediction games and arcing algorithms. Neural Comput 11(7):1493–1517. https://doi.org/10.1162/089976699300016106

Brown G, Wyatt JL, Tiño P (2005a) Managing diversity in regression examples. J Mach Learn Res 6:1621–1650

Brown G, Wyatt JL, Harris R, Yao X (2005b) Diversity creation methods: a survey and categorization. Information Fusion 6:5–20

Brown JB, Salehi A, Benhallam W, Matringe S (2017) Using data-driven technologies to accelerate the field development planning process for mature-field rejuvenation. SPE Western Regional Meeting, Bakersfield, California, USA, 23–27 April. SPE 185751

Cawley GC, Talbot NLC (2010) On over-fitting in model selection and subsequent selection bias in performance evaluation. J Mach Learn Res 11:2079–2107

Chen T, Guestrin C (2016) XGBoost: a scalable tree boosting system. KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 785–794. https://doi.org/10.1145/2939672.2939785

Chen D, Zhong A, Gano J, Hamid S, De Jesus O, and Stephenson S (2007) Construction of surrogate model ensembles with sparse data. IEEE Congress on Evolutionary Computation, Singapore 244–251. doi: https://doi.org/10.1109/CEC.2007.4424478

Chen L, Qiu H, Jiang C, Cai X, Gao L (2018) Ensemble of surrogates with hybrid method using global and local measures for engineering design. Struct Multidisc Optim 57(4):1711–1729. https://doi.org/10.1007/s00158-017-1841-y

Cherkassky V, Ma Y (2004) Practical selection of SVM parameters and noise estimation for SVM regression. Neural Netw 17:113–126. https://doi.org/10.1016/S0893-6080(03)00169-2

Dixon LCW, Szego GP (1978) The global optimization problem: an introduction. Towards global optimization 2:1–15

Dixon LCW, Price RC (1989) The truncated Newton method for sparse unconstrained optimisation using automatic differentiation. J Optim Theory Appl 60(2):261–275. https://doi.org/10.1007/BF00940007

Draper N, Smith H (1969) Applied regression analysis. Biom J 11(6): 427–427. https://doi.org/10.1002/bimj.19690110613

Fang J, Gao Y, Sun G (2014) Fatigue optimization with combined ensembles of surrogate modeling for a truck cab. J Mech Sci Technol 28:4641–4649

Fayyad UM, Irani KB (1992) On the handling of continuous-valued attributes in decision tree generation. Mach Learn (1992) 8:87–102. https://doi.org/10.1007/BF00994007

Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189–1232

Freund Y (1995) Boosting a weak learning algorithm by majority. Inf Comput 121(2):256–285. https://doi.org/10.1006/inco.1995.1136

Goel T, Haftka RT, Shyy W, Queipo NV (2007) Ensemble of surrogates. Struct Multidisc Optim 33(3):199–216

Hamza K, Saitou K (2012) A co-evolutionary approach for design optimization via ensembles of surrogates with application to vehicle crashworthiness. Transactions of ASME Journal of Mechanical Design 134(1):011001–1–011001-10

Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning. Springer Series in Statistics, New York

Hecht-Nielsen R (1989), Neurocomputing. Addison Wesley, Boston, USA

Ho TK (1998) The random subspace method for constructing decision forests. IEEE PAMI 20(8):832–844. https://doi.org/10.1109/34.709601

Karagiannopoulos M, Anyfantis D, Kotsiantis S, Pintelas P (2007) Feature selection for regression problems. In: Proceedings of the 8th Hellenic European Research on Computer Mathematics and its Applications (HERCMA). Greece, Athens

Kazemitabar J, Amini A, Bloniarz A, Talwalkar AS (2017) Variable importance using decision trees. In: Advances in neural information processing systems, vol 30, pp 426–435

Liu Y, Yao X (1999) Ensemble learning via negative correlation. Neural Netw 12(10):1399–1404. https://doi.org/10.1016/S0893-6080(99)00073-8

Mendes-Moreira J, Soares C, Jorge AM, De Sousa JF (2012) Ensemble approaches for regression: a survey. ACM Comput Surv 45(1):1–40. https://doi.org/10.1145/2379776.2379786

Poggio T, Smale S (2003) The mathematics of learning: dealing with data. Not Am Math Soc 50(5):537–544

Reeve HWJ, Brown G (2018) Diversity and degrees of freedom in regression ensembles. Neurocomputing 298(12):55–68

Romero J, Queipo N (2017) Reliability-based and deterministic design optimization of a FSAE brake pedal: a risk allocation analysis. Struct Multidisc Optim 56(3):681–695. https://doi.org/10.1007/s00158-017-1747-8

Rooney N, Patterson D, Anand S, Tsymbal A (2004) Dynamic integration of regression models. In: Roli F, Kittler J, Windeatt T (eds). Multiple classifier systems. MCS 2004. Lect Notes Comput Sci 3077:164–173. https://doi.org/10.1007/978-3-540-25966-4_16

Rosen BE (1996) Ensemble learning using decorrelated neural networks. Connect Sci 8(3–4):373–384. https://doi.org/10.1080/095400996116820

Rosenbrock HH (1960) An automatic method for finding the greatest or least value of a function. Comput J 3(3):175–184. https://doi.org/10.1093/comjnl/3.3.175

Sacks J, Welch W, Mitchell T, Wynn H (1989) Design and analysis of computer experiments. Stat Sci 4(4):409–435

SAE International (2016a) SAE Collegiate Design Series http://students.sae.org/cds/

SAE International (2016b) Formula SAE Events History http://students.sae.org/cds/formulaseries/history/

SAE International (2016c) About Formula SAE® Series http://students.sae.org/cds/formulaseries/about/

Sanchez E, Pintos S, Queipo NV (2008) Toward an optimal ensemble of kernel-based approximations with engineering applications. Struct Multi Optim 36(3):247–261

Schapire RE (1990) The strength of weak learnability. Mach Learn 5(2): 197–227. https://doi.org/10.1007/BF00116037

Schölkopf B, Smola AJ (2002) Learning with kernels. MIT Press, Cambridge, MA

Sobol I (1993) Sensitivity analysis for non-linear mathematical models. Mathematical Modeling & Computational Experiment 1:407–414

Syed A (2012) Technology focus: mature fields and well revitalization. J Pet Technol 64:74–74. https://doi.org/10.2118/0112-0074-JPT

Tenne Y (2013) An optimization algorithm employing multiple metamodels and optimizers. Int J Autom Comput 10(3):227–241. https://doi.org/10.1007/s11633-013-0716-y

Ueda N and Nakano R (1996). Generalization error of ensemble estimators. In Proceedings of International Conference on Neural Networks(ICNN'96), 90–95. https://doi.org/10.1109/ICNN.1996.548872

Viana FAC, Simpson TW, Balabanov V, Toropov V (2014) Metamodeling in multidisciplinary design optimization: how far have we really come? AIAA J 52(4):670–690. https://doi.org/10.2514/1.J052375

Yin H, Fang H, Wen G, Gutowski M, Xiao Y (2018) On the ensemble of metamodels with multiple regional optimized weight factors. Struct Multidisc Optim 58:245–263. https://doi.org/10.1007/s00158-017-1891-1

Zerpa L, Queipo NV, Pintos S, Salager JL (2005) An optimization methodology of alkaline–surfactant–polymer flooding processes using field scale numerical simulation and multiple surrogates. J Pet Sci Eng 47(3–4):197–208. https://doi.org/10.1016/j.petrol.2005.03.002

Zhang J, Chowdhury S, Messac A (2012) An adaptive hybrid surrogate model. Struct Multi Optim 46(2):223–238. https://doi.org/10.1007/s00158-012-0764-x