Struct Multidisc Optim (2011) 44:651–671
DOI 10.1007/s00158-011-0655-6

RESEARCH PAPER

Ensemble of surrogates with recursive arithmetic average

Xiao Jian Zhou · Yi Zhong Ma · Xu Fang Li

Received: 30 July 2010 / Revised: 31 March 2011 / Accepted: 4 April 2011 / Published online: 7 May 2011

Abstract Surrogate models are often used to replace expensive simulations of engineering problems. The common approach is to construct a series of metamodels based on a training set, and then, from these surrogates, pick out the best one with the highest accuracy as an approximation of the computationally intensive simulation. However, because the choice of approximate model depends on design of experiments (DOEs), the traditional strategy thus increases the risk of adopting an inappropriate model. Furthermore, in the design of complex product system, because of its feature of one-of-a-kind production, acquiring more samples is very expensive and intensively time-consuming, and sometimes even impossible. Therefore, in order to save sampling cost, it is a reasonable strategy to take full advantage of all the stand-alone surrogates and then combine them into an ensemble model. Ensemble technique is an effective way to make up for the shortfalls of traditional strategy. Motivated by the previous research on ensemble of surrogates, a new technique for constructing of a more accurate ensemble of surrogates is proposed in this paper. The weights are obtained using a recursive process, in which the values of these weights are updated in each iteration until the last ensemble achieves a desirable prediction accuracy. This technique has been evaluated using five benchmark problems and one reality problem. The results show that the proposed ensemble of surrogates with recursive arithmetic average provides more ideal prediction accuracy than the stand-alone surrogates and for most problems even exceeds the previously presented ensemble techniques. Finally, we should point out that the advantages of combination over selection are still difficult to illuminate. We are still using an "insurance policy" mode rather than offering significant improvements.

Keywords Metamodel · Surrogate · Ensemble · Design of experiment · Recursive arithmetic average

1 Introduction

With the continuing updating of CPU and escalation of memory, the computer processing power has drastically increased, but the computational cost of complex high-fidelity engineering simulations often makes it impractical to rely exclusively on simulation for design optimization (Jin et al. 2001). Just taking Ford Motor Company as an example, it reported that it takes the company about 36–160 h to run one crash simulation (Wang and Shan 2007). For a two-dimension optimization problem, assuming that on average 50 iterations are needed in the optimization process, and assuming that each iteration requires one crash simulation, then the total amount of computation time would reach to as much as 75 days to 11 months, which is unacceptable

Part of the work was presented at the 2010 2nd International Conference on Industrial Mechatronics and Automation (ICIMA 2010), Wuhan, China.

X. J. Zhou (✉) · Y. Z. Ma
Department of Management Science and Technology,
Nanjing University of Science and Technology,
Nanjing, China
e-mail: xjzhou2008@yahoo.com.cn

Y. Z. Ma
e-mail: yzma-2004@163.com

X. F. Li
Temasek Laboratories, Nanyang Technological University,
50 Nanyang Ave., Singapore 639798, Singapore
e-mail: lixufang@ntu.edu.sg

in practice. In order to reduce the computational cost, surrogate models (also referred to as "metamodels") are used to replace the expensive simulation models (Queipo et al. 2005; Viana et al. 2010). Surrogate evolves from the classical Design of Experiments (DOE) theory, in which the polynomial model is known as "response surface model". Essentially, it is also a kind of surrogate. In addition to commonly used polynomial model, Sacks et al. (1989a, b) proposed a stochastic model, i.e., Kriging (Cresssie 1988), to treat the deterministic computer response as a realization of a random function with respect to the actual system response. Neural networks are also often applied to simulate the responses for complex systems (Papadrakakis et al. 1998). Other types of metamodels include radial basis functions (RBF) (Fang and Horstemeyer 2006), multivariate adaptive regression splines (MARS) (Friedman 1991), least interpolating polynomials (De Boor and Ron 1990), inductive learning (Langley and Simon 1995), support vector regression (SVR), and so on. In general, Kriging model is more accurate for non-linear problems than other models due to its capacity of interpolating the sample points and filtering noisy data, but it is difficult to be obtained and used because a global optimization process is involved to identify the maximum likelihood estimators. In contrary to Kriging model, polynomial models are relatively easy to be built up and clear on parameter sensitivity but unsatisfactory in accuracy because of the difficulty in determining its model structure (its highest order and the number of items) (Jin et al. 2001). The RBF model, particularly the multi-quadric RBF, can interpolate sample points and is easy to build, which thus seems to reach a trade-off between Kriging models and polynomial models. SVR has been intensively studied in the area of machine learning but seldom used in computer experiment. Its capacity of fitting of data has been tested and verified in Clarke et al. (2005), which shows that the higher accuracy was achieved, compared with all other metamodeling techniques including Kriging, polynomials, RBF and MARS in a series of test problems. Just as the author pointed out, the basic reasons why SVR outperforms others are not clear. More recent and comprehensive reviews of metamodeling can be traced to Kleijnen et al. (2005), Wang and Shan (2007), Simpson et al. (2008) and Forrester and Keane (2009).

If only one single predictor is desired, there are two strategies for us to obtain the final prediction surrogate. One is selection, which can be done using cross validation (Picard and Cook 1984; Kohavi 1995); the other is combination, which can be traced to the development of committees of neural networks by Perrone and Cooper (1993) with further refinement by Bishop (1995). Zerpa et al. (2005) and Goel et al. (2007) extended this idea to the ensemble of metamodels. Goel et al. (2007) found that multiple metamodels can be used to identify the regions of possible high errors where predictions of metamodels differ widely. Thereby this can guide the engineer to gather more sample points in this uncertain region to achieve more accurate result. In addition, the authors also found that combining of metamodels can provide us with a more robust ensemble, which can effectively eliminate the negative impact brought by inappropriate stand-alone metamodel, that is, the use of multiple surrogates acts like an insurance policy against poorly fitted models, which is also confirmed by Viana et al. (2009). Acar and Rais-Rohani (2009) proposed a combining technique with optimized weight coefficients, which are obtained by solving an optimization problem. The technique in Acar and Rais-Rohani (2009) could achieve a certain satisfactory result in some cases, nevertheless, it has several deficiencies as following: (1) The optimization problem used to determine the weight coefficients could not ensure obtaining a global optimal solution, and is easily trapped into a local optimum, and even has no local optimal solution; and (2) The range of weight coefficients are not constrained to $w_i \geq 0$ when solving the optimization problem, as $w_i < 0$ is difficult to be explained in actual problems. In Acar and Rais-Rohani (2009), authors get the weights by minimizing GMSE or RMSE^v using a formal optimization algorithm in MATLAB. In terms of minimizing RMSE^v, the technique is essentially the same as the Bishop's approach on minimizing the mean square error (MSE). Inspired by the works of Bishop (1995) and Acar and Rais-Rohani (2009), Viana et al. (2009) also obtained the weight coefficients by minimizing MSE. Viana et al. (2009) got the solution of the weight via Lagrange multipliers, and the authors replaced the real error covariance matrix C with cross-validation error matrix, with the corresponding method named OWS (optimal weighted surrogate) in the literature. However, OWS is essentially the same as the approach based on minimizing GMSE in Acar and Rais-Rohani (2009). In order to make the solution range between zero and one, Viana et al. (2009) only used the diagonal elements of C, with the corresponding method named OWS_{diag} in the literature, and just as the authors said in their paper, this method has similar structure and prediction accuracy to the approach named heuristic computation of the weights in Goel et al. (2007). In addition to these ensemble techniques mentioned above, there are several other ensemble techniques appeared in the literatures, such as BestPRESS (Goel et al. 2007), OWS_{ideal} Viana et al. (2009), and so on. Essentially, OWS_{ideal} Viana et al. (2009) is the same as minimizing RMSE^v in Acar and Rais-Rohani (2009). The difference between them is that RMSE^v in Acar and Rais-Rohani (2009) employs a formal optimization algorithm, while OWS_{ideal} Viana et al. (2009) is obtained via Lagrange multipliers.

Motivated by the existing works, the ensemble technique with recursive arithmetic average is proposed in this paper.

The weights are obtained using a recursive process, in which the values of these weights are updated in each iteration until the last ensemble reach to a desirable prediction accuracy. This technique builds an ensemble of metamodels by recursive arithmetic average several times rather than arithmetically averaging the responses of the stand-alone metamodels just once. In order to illustrate the performance of the proposed technique, four types of metamodeling techniques (polynomial function, Kriging, RBF and SVR) are used to build up the ensemble, and these four stand-alone metamoels as well as the existing ensemble techniques are compared with the ensemble technique proposed in this paper. The performances of these stand-alone metamodels and all of the ensembles are evaluated by several commonly used criteria (e.g., correlation (denoted by R), maximum absolute error (MAE), average absolute error (AAE), root of mean square error (RMSE), etc.). The experimental results showed that the proposed ensemble of metamodels with recursive arithmetic average provides more accurate predictions than the stand-alone metamodels and for most problems even exceeds the previously presented ensemble techniques.

The remainder of this paper is organized as follows. In the next section, we present the basic weighted-sum formulation and the different techniques that can be used to select the weight factors for the stand-alone metamodels. In Section 3, the test problems are considered and the numerical procedure for finding an ensemble with recursive arithmetic average is presented. The presentation and discussion of results is displayed in Section 4. At last, the summary of several important conclusions is discussed in Section 5.

2 Ensemble of surrogates

For a given problem, if all the candidate metamodels developed for a given high-fidelity simulation happen to have the same level of accuracy, then a very straightforward form for the ensemble would be a simple average of the surrogates. However, for a specified problem the usual case is that there are some models that are more accurate than others. Therefore, in order to improve the accuracy of ensemble, the stand-alone surrogates have to be multiplied by different weight coefficients. Using the weight-sum formulation, the ensemble of surrogates for approximation of response can be expressed as:

$$\widehat{y}_s(x) = \sum_{i=1}^{N} w_i(x)\widehat{y}_i(x) \sum_{i=1}^{N} w_i(x) = 1 \tag{1}$$

where x is input variable, $\widehat{y}_s(x)$ is the ensemble response, N is the number of surrogates in the ensembles, $w_i(x)$ is the weight coefficient for the ith surrogate, $\widehat{y}_i(x)$ is the response estimated by the ith surrogate.

Generally, the weight coefficients are selected such that the surrogates with high accuracy have large weight factor and vice versa.

All of the ensembles of surrogates in literatures can be divided into three categories:

(1) Combining surrogates by minimizing cross-validation errors (GMSE; PRESS in particular), e.g., heuristic computation of the weight coefficient (Goel et al. 2007), the approach based on minimizing GMSE^v in Acar and Rais-Rohani (2009), OWS, OWS$_{diag}$ (Viana et al. 2009), and BestPRESS (Goel et al. 2007; Viana et al. 2009);
(2) Combining surrogates using prediction variance, e.g., the approach obtaining the weights based on variance reciprocal (Bishop 1995; Zerpa et al. 2005);
(3) Combining surrogates by minimizing mean square error (or root of mean square error (RMSE)), e.g., OWS$_{ideal}$ (Viana et al. 2009), the approach based on minimizing RMSE^v in Acar and Rais-Rohani (2009).

In the first category, the weights are determined using training points, but, in the second and third category, the weight is determined using several validation points in test set. The techniques determining the weights using cross validation are time-consuming, while the ones using validation points all require additional simulations for response determination. Depending on the type of surrogate and the computational cost of simulation calculation, one error metric (PRESS or RSME) would be less expensive to evaluate than the others (PRESS or RSME). If the cost of obtaining data required for developing surrogate models is high, choosing PRESS as error metric would be a reasonable strategy, for additional response validations at test set are needed with RMSE. On the contrary, if the surrogate-constructing is computationally costly, RMSE (or MSE) used as error metric would be a better choice, for only a single surrogate would be constructed with RMSE. The technique proposed in this paper belongs to the third category. Next, the details of all the ensembles are presented below.

2.1 Weight coefficients selection based on prediction variance

Based on the work of Bishop (1995), Zerpa et al. (2005) used the ensemble of surrogates including response surface (RS) model, Kriging model and RBF model in the optimization of an alkali-surfactant polymer flooding process, and chose the prediction variance as the error metric. The values

of the weight coefficients are determined by the following formula:

$$w_i = w_i^* \Big/ \sum_{i=1}^{M} w_i^*, \; w_i^* = \frac{1}{V_i} \tag{2}$$

where V_i is the prediction variance of the ith surrogate.

2.2 Combining surrogates by minimizing cross-validation errors

2.2.1 Heuristic computation of the weight coefficient

Goel et al. (2007) proposed a heuristic method for calculating the weight coefficients, which is known as PRESS (predicted residual sum of squares) weighted average surrogate, where the weight coefficients are computed as:

$$w_i = w_i^* \Big/ \sum_{i=1}^{M} w_i^*, \; w_i^* = (E_i + \alpha E_{avg})^{\beta},$$
$$E_{avg} = \frac{1}{n}\sum_{i=1}^{n} E_i, \beta < 0, \alpha < 1 \tag{3}$$

where E_i is the PRESS error of the ith surrogate, α, β are used to control the importance of averaging and individual PRESS respectively. Goel et al. (2007) suggested $\alpha = 0.05, \beta = -1$.

2.2.2 The approach based on minimizing GMSE^v

Acar and Rais-Rohani (2009) proposed a method for determining the weight coefficients, which is achieved through minimizing some error metric, such as PRESS error. The optimization problem is presented as:

$$\min \varepsilon_s = Err\left\{\widehat{y}_s\left(w_i, \widehat{y}_i(\mathbf{x}^k)\right) y_i\left(\mathbf{x}^k\right), k = 1 \in N\right\}$$
$$s.t. \sum_{i=1}^{N} w_i = 1 \tag{4}$$

where $Err\{\cdot\}$ is the selected error metric which measures the accuracy of the ensemble-predicted response $\widehat{y}_s$. The author adopted the generalized mean square cross-validation error (GMSE; leave-one-out cross validation or PRESS in polynomial response surface approximation terminology) as one kind of the error metric.

2.2.3 OWS (Optimal weighted surrogate)

Employing an ensemble of neural networks, Bishop (1995) proposed a weighted surrogate obtained by approximating the covariance between surrogates from residuals at training or test points, whose approach is based on mimizing the MSE:

$$\mathrm{MSE}_{\mathrm{WAS}} = \frac{1}{\mathrm{V}}\int_V e_{\mathrm{WAS}}^2(\mathbf{x})d\mathbf{x} = \mathbf{w}^T C\mathbf{w} \tag{5}$$

where $e_{\mathrm{WAS}}(\mathbf{x}) = y(\mathbf{x}) - y_{\mathrm{WAS}}(\mathbf{x})$ is the error associated with the prediction of the WAS ensemble model, and the integral, which is taken over the domain interest, permits the calculation of the elements of C as:

$$c_{ij} = \frac{1}{V}\int_V e_i(\mathbf{x})e_j(\mathbf{x})d\mathbf{x} \tag{6}$$

where $e_i(\mathbf{x})$ and $e_j(\mathbf{x})$ are the errors associated with the prediction given by the surrogate model i and j respectively.

C plays the same role as the the covariance matrix in Bishop's formulation. But C is approximated by the vectors of cross validation errors, $\tilde{e}$,

$$c_{ij} \simeq \frac{1}{p}\tilde{e}_i^T\tilde{e}_j \tag{7}$$

where p is the number of data points and the i and j indicate different surrogates.

Given the C matrix, the optimal weighted surrogate (OWS) is obtained by minimizing the MSE as:

$$\min_{\mathrm{w}} \mathrm{MSE}_{\mathrm{WAS}} = \mathbf{w}^T\mathbf{C}\mathbf{w} \tag{8}$$

s.t. $\mathbf{1}^T\mathbf{w} = 1$.

Using Lagrange multipliers, the solution is obtained as:

$$\mathbf{w} = \frac{\mathbf{C}^{-1}\mathbf{1}}{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{1}} \tag{9}$$

The weight in the formulation above may less than zero or larger than one, whose meaning is difficult to explain in real world, and, as pointed out by Viana et al. (2009), allowing this freedom was found to amplify errors coming from the approximation of matrix (7). In Viana et al. (2009), the author enforced the weight positive by solving (9) using only the diagonal elements of C. The approach is named OWS_{diag}.

After examining formulas (4) and (9), we can find that both approaches actually the same, for both of them are all based on minimizing cross validation (especially PRESS; GMSE). The difference between them is that the approach in Acar and Rais-Rohani (2009) obtains the weights through a optimization process, while the approach in Viana et al. (2009) obtains the weights through an analysis expression, however, both approaches have exactly the same solution. Thereby, in order to avoid replication, OWS is not included in the rest of this paper.

2.2.4 BestPRESS

The traditional method of using an ensemble of surrogates is to select the best surrogate among all of the considered models. However, once the choice is made, the surrogate is fixed even though the design of experiments is changed. If the choice is refined for each new DOE, we can included it in the strategies for multiple surrogates, where the model with least error is assigned a weight of one and all others are assigned zero weight. Just as many literatures do, we also call this strategy BestPRESS model.

2.3 Combining surrogates by minimizing mean square error (MSE) (or root of mean square error (RMSE))

2.3.1 OWS_{ideal}: the approach based on minimizing $RMSE^{v}$ in Acar and Rais-Rohani (2009)

In formula (7), if $\tilde{e}$ is the real MSE in validation point set rather than the cross-validation in training set, then C is not the cross-validation error covariance matrix but the real error covariance matrix in formula (9). Just as we have pointed out above, OWS_{ideal} is exactly the same as the approach based on minimizing RMSE^{v} in Acar and Rais-Rohani (2009). In Acar and Rais-Rohani (2009), the RMSE^{v} (where v is number of validation points in test set) is chosen as the error metric in formula (4). Therefore, in order to avoid replication, the remainder of this paper doesn't include OWS_{ideal}.

2.3.2 The strategy proposed in this paper-ensemble of surrogates with recursive arithmetic average

As having been mentioned above, most of the ensemble techniques obtain the weights by either minimizing cross-validation errors or minimizing RMSE (or MSE). Although the techniques using cross-validation errors don't require additional validation points, they must be constructed many times, thereby, they are time-consuming. On the contrary, the techniques with RMSE (or MSE) need additional validation points, but these approaches only need to construct the surrogates once, so they are time-saving. In addition, when the value of RMSE at the test points is used as the error criterion, the techniques using RSME usually have better results, for the error metric employed in obtaining the weights is the same as that in measuring the prediction accuracy (they all use RMSE). The technique proposed in this paper also employs the prediction mean square error as the error metric.

In all of the combining techniques, the simplest and straight forward approach is to arithmetically average these single surrogates. Nevertheless, arithmetically averaging the stand-alone surrogates just once would not minimize the prediction mean square error. In order to make the prediction mean square error as low as possible, we consider to employ recursive process. Generally, the iteration in recursive process should be repeated several times, how many of which depends on the specified stop criterion. In this strategy, the algorithm stops when the prediction MSE of the worst surrogate approaches to that of the best surrogate. In other words, all the updated surrogates in the last iteration have similar prediction results (i.e., similar prediction MSEs). Furthermore, we should point out that the surrogates in the recursive process are not the initial single surrogates but the combining surrogates obtained using arithmetically averaging. The basic frame of this algorithm is as follows:

Input: Initial weight coefficients

Step 0: Fit the training data $\{x_j\}$, $j = 1, 2, ..., T$ (where T is the number of the training points) with N candidate surrogates;

Step 1: Calculate their prediction mean square errors: $e_i = \frac{1}{T}\sum_{j=1}^{T}(Sur_{ij} - \widehat{Sur}_{ij})^2, i = 1, 2, ..., N$ (where $\widehat{Sur}_{ij}$ is the prediction value on the jth validation point of the ith individual surrogate) on the validation points;

Step 2: Find out the worst individual surrogate (i.e., the surrogate that has the largest prediction MSE, denoted by Sur_{worst}, and its corresponding prediction MSE is denoted by $\text{MSE}_{WorstSur}$) and the best surrogate (i.e., the surrogate that has the smallest prediction MSE, denoted by Sur_{best}, and its corresponding prediction MSE is denoted by $\text{MSE}_{BestSur}$).
While($\text{MSE}_{WorstSur} - \text{MSE}_{BestSur} > tol$)D

Step 3: Obtain the arithmetic average of the candidate N surrogates; that is, all the candidate single surrogates are added, and then divided by the total number of all the candidate surrogates; denote this average ensemble model using Sur_{ave};

Step 4: Replace the surrogate which has the largest prediction MSE (i.e. Sur_{worst}) with the simple average surrogate (i.e. Sur_{ave}) made in step 3 (this surrogate replaced by average surrogate may be one of the initial candidate surrogates or the average ensemble model in the previous time), then we can get N new surrogates, of which $N - 1$ surrogates are not changed; calculate and then update the weights for the initial individual surrogates;

Step 5: Do the same work as that in step 2; if the condition in while $(\cdot)$ is met, then return to step 3, otherwise break out of the loop.
EndWhile

Output: Optimal weight cofficients

Such iteration will be taken until the prediction MSE has no significant improvement. In the algorithm above, tol is the tolerant value determined in advance (e.g., $tol = 0.01$). Next, the convergence of the above-mentioned algorithm is presented as follows.

For a problem, there are N kinds of surrogates Sur_1, Sur_2, ..., Sur_N, the weight for Sur_i is w_i, and $\sum_{i=1}^{N} w_i = 1$. Assume the prediction value and prediction error of the ith surrogate Sur_i on the jth data point respectively are Sur_{ij} and e_{ij}, $j = 1, 2, ..., T$ (where T is the number of the training points), then the prediction value and prediction error of the simple average surrogate Sur_{ave} on the jth data point respectively are $Sur_{ave}(j) = \sum_{i=1}^{N} w_i Sur_{ij}$ and $e_{ave}(j) = \sum_{i=1}^{N} w_i e_{ij}$. Denote the weight vector by $W = [w_1, w_2, ...w_N]^T$, denote the prediction error vector of the Sur_i by $E_i = [e_{i1}, e_{i2}, ...e_{iT}]^T$, denote the prediction error matrix by $e = [E_1, E_2, ...E_N]$, and denote the sum of prediction square error of the simple average surrogate by J, then the following stands:

$$J = W^T E W, \tag{10}$$

where

$$E = e^T e = \begin{bmatrix} E_{11} & E_{12} & \dots & E_{1N} \\ E_{21} & E_{22} & \dots & E_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ E_{N1} & E_{N2} & \dots & E_{NN} \end{bmatrix},$$

and where

$$E_{ij} = E_i^T E_j = \sum_{i=1}^{N} e_{it} e_{jt}.$$

Apparently, E_{ii} is the sum of prediction square error of Sur_i.

Based on the description above, we have the following lemma.

Lemma 1 *Assume the prediction error vector $E_1, E_2, ..., E_N$ is linear independent, and denote the sum of prediction square error of the simple average surrogate by J_A, then*

$$J_A < J_{\max}. \tag{11}$$

Proof The weights of the simple average surrogate is

$$W_A = [1/N, 1/N, ...1/N]^T, \tag{12}$$

and

$$J_A = W_A^T E W_A = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{t=1}^{T} e_{it} e_{jt}. \tag{13}$$

Because $E_1, E_2, ..., E_N$ is linear independent, then

$$\sum_{t=1}^{N} e_{it} e_{jt} < \sqrt{\sum_{t=1}^{N} e_{it}^2} \sqrt{\sum_{t=1}^{N} e_{jt}^2} = \sqrt{E_{ii}} \sqrt{E_{jj}} \le \sqrt{J_{\max}} \sqrt{J_{\max}} = J_{\max}, \tag{14}$$

so,

$$J_A < \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{\max} = J_{\max}.$$

The proof is finished. □

Theorem 1 *Denote the error vector which is obtained by replacing the worst surrogate with the simple average surrogate (i.e.* Sur_{ave}*) in kth iteration by*

$$E^{(k)} = \left(E_{11}^{(k)}, E_{22}^{(k)}, ...E_{NN}^{(k)} \right), \tag{15}$$

then

$$\lim_{k \to \infty} E^{(k)} = (d, d, ...d), \tag{16}$$

where, $d = MSE_{BestSur}$.

Proof Denote $E_{\max}^{(0)} = \max\{E_{ii}\}$ and $E_{\max}^{(k)} = \max\{E_{ii}^{(k)}\}$, where $i = 1, 2, ...N$. Because the worst surrogate is replaced by the simple average surrogate in each iteration, according to lemma 1, $E_{\max}^{(0)} > E_{\max}^{(1)} > ... > E_{\max}^{(k)} > ....$ On the other hand, the best initial surrogate is not changed in each iteration, then $E_{\max}^{(k)} \ge MSE_{BestSur}$. Because it is monotonous and bounded, the data serial $\left\{ E_{\max}^{(k)} \right\}_{k=0}^{\infty}$ has its limit, denoted by d, i.e., $\lim_{k \to \infty} E_{\max}^{(k)} = d$.

Apparently, $d \ge MSE_{BestSur}$. Next, we will prove $d = MSE_{BestSur}$. In fact, if $d > MSE_{BestSur}$, according to lemma 1, we can replace the worst surrogate with the simple average surrogate, then the prediction MSE of the worst surrogate will less than d in the next iteration, which is contradict to the conclusion $\lim_{k \to \infty} E_{\max}^{(k)} = d$. Therefore, $d = MSE_{BestSur}$, i.e., $\lim_{k \to \infty} E_{max}^{(k)} = MSE_{BestSur}$.

Furthermore, denote $E_{\min}^{(0)} = \min\{E_{ii}\}$ and $E_{\min}^{(k)} = \min\{E_{ii}^{(k)}\}$, we can easily know $E_{\min}^{(0)} = E_{\min}^{(1)} = ... = E_{\min}^{(k)} = ... = MSE_{BestSur}$. So, $\lim_{k \to \infty} E^{(k)} = (d, d, ...d)$, where $d = MSE_{BestSur}$. The proof is finished. □

The technique proposed in this paper has several differences from the existing ensemble techniques:

(1) Because cross-validation often tends to overestimate errors, the real gain in accuracy of the ensemble technique based on cross-validation is limited, the illustration about which is presented in Viana et al. (2009). However, as for the third class of ensemble technique based on minimizing RMSE mentioned above, if the validation points are acquired easily, we can consider to get more validation points to construct the ensemble. Generally, the more validation points are used to determine the weights in ensemble of surrogates, the better prediction accuracy can be achieved by the ensemble. If the validation point set is large, the prediction MSE of the ensemble of surrogates would approache to that of the BestRMSE (Viana et al. 2009). In the process of obtaining the weights, the validation points are also needed in the technique proposed here, and with recursive scheme, the proposed technique can achieve desirable results. In a word, the technique proposed in this paper is based on minimizing RMSE, and, because it adopt recursive process, has an ideal prediction capacity, which is the difference of the proposed technique in this paper from those techniques based on minimizing cross-validation (especially, GMSE; PRESS).

(2) As for OWS_{ideal} (Bishop 1995; Viana et al. 2009), using Lagrange multipliers to get the weight solution can neither ensure the weights larger than or equal to one nor ensure not less than zero, whose physical meaning in many circumstances is difficult to explain. Similarly, the approach based on minimizing RMSE^v (Acar and Rais-Rohani 2009) also hasn't added the condition $w_i \geq 0$ into formula (4). If $w_i \geq 0$ is added into formula (4), the analysis expression like (9) cannot been obtained, and a lot of iterations in simplex method of operational research or other formal intelligent optimization algorithm would be needed. Thereby, when the dimension of the problem is large, the optimization process is also time-consuming. So, a simple and straight-forward approach is needed. Arithmetic average ensemble surrogate proposed in this paper can ensure the weights nonnegative and not larger than one, which is convenient to explain the importance of each candidate single surrogate.

(3) As mentioned in (2), the optimization process is also time-consuming, especially in problems with large dimensions. On the contrary, recursive process is time-saving compared to optimization process. The number of iterations is effected by tol and usually is a dozen or dozens, so it executes more quickly than optimization process. The experiment results presented in the end of Section 4 confirm this.

3 Experiments

3.1 Benchmark problems

In order to test the proposed technique in this paper, we choose the following analytic functions that are commonly used as benchmark problems in literatures.

Branin–Hoo:

$$y(x_1, x_2) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10 \tag{17}$$

where $x_1 \in [-5, 10]$, $x_2 \in [0, 15]$.

CamelBack:

$$y(x_1, x_2) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(-4 + 4x_2^2\right)x_2^2 \tag{18}$$

where $x_1 \in [-3, 3]$, $x_2 \in [-2, 2]$.

Goldstein–Price:

$$\begin{aligned} y(x_1, x_2) = {} & \left[1 + (x_1 + x_2 + 1)^2 \times \left(19 - 4x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right] \\ & \times \left[30 + (2x_1 - 3x_2)^2 \times \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right] \end{aligned} \tag{19}$$

where $x_1, x_2 \in [-2, 2]$.

Hartman:

$$y(\mathbf{x}) = -\sum_{i=1}^{m} c_i \exp\left[-\sum_{j=1}^{n} a_{ij}(x_j - p_{ij})^2\right] \tag{20}$$

where $x_i \in [0, 1]$.

Both the three-variables ($n = 3$) and the six-variables ($n = 6$) models of this function are considered. The values of function parameters c_i, p_{ij}, a_{ij} for Hartman-3 and Hartman-6 models, given in Tables 1 and 2, are taken from Goel et al. (2007) and Acar and Rais-Rohani (2009). For the chosen examples, $m = 4$.

Table 1 Parameters used in Hartman function with three variables

i	a_{ij}			c_i	p_{ij}		
1	3.0	10.0	30.0	1.0	0.3689	0.1170	0.2673
2	0.1	10.0	35.0	1.2	0.4699	0.4387	0.7470
3	3.0	10.0	30.0	3.0	0.1091	0.8732	0.5547
4	0.1	10.0	35.0	3.2	0.03815	0.5743	0.8828

3.2 Abalone problems

In the prediction of the life-span of abalone, every sample of abalone includes the following eight indicators: sex, length, diameter, thickness, total weight, the weight apart from shell, the weight of guts, and the weight of shell. The life-span of abalone is predicted according to the above-mentioned indicators. We choose 200 samples for this experiment from http://archive.ics.uci.edu/ml/datasets/Abalone.

3.3 Design and analysis of computer experiments

As for these five test functions presented in formulas (17)–(20) and the Abalone problem, all of them use the Latin hypercube sampling (LHS). Some people also call it the symmetrical LHS sample to distinguish from Latin hypercube(LH), which keeps the mid-point principle. These kinds of sampling have a better nature than Monte-Carlo sampling (or call it simple random sampling). In this paper we have adopted the principle of maximizing the minimum distance, which refers to finding the set of sample that meets the formula $\max\{\min_{i\neq j} d(x_i, x_j)\}$ (where d is some kind of criterion to measure distance) in n ($n = 20$ in benchmark problems and $n = 80$ in Abalone problem) times repeated sampling.

In order to reduce the influence of random factors, we randomly select 1,000 training sets for these three test functions expressed in formulas (17)–(19) and the Hartman-3. However, considering the computational cost, we select 200 training sets for Hartman-6 and 500 ones for Abalone. Depending on the number of input variables, and considering the computational cost, the training set for each benchmark problem is composed of 12–60 design points, which are the same as that in Acar and Rais-Rohani (2009). For these ensembles which depend on minimizing RMSE (or prediction MSE), there are additional validation points needed. Depending on the precision level sought for estimating the error, the number of validation points, denoted by V, will vary with different problem. $V = 0.8N$ (where N is the no. of training points) was used in these approaches based on minimizing RMSE (certainly including the technique proposed in this paper). Hence, all the corresponding surrogates, including stand-alone surrogates and ensembles, are constructed multiple times with the error estimation being the average value corresponding to multiple replication of the same surrogate. Additional information about the training and test data sets is provided in Table 3.

Table 3 Summary of training and test data used in each benchmark problem

Benchmark problem	Design variables	Training sets	Design point	Test point
Branin–Hoo	2	1,000	12	441
Camelback	2	1,000	12	441
Goldstein–Price	2	1,000	12	441
Hartman-3	3	1,000	20	441
Hartman-6	6	200	56	512
Abalone	8	500	60	140

The accuracies of each stand-alone and ensemble model for the benchmark problems are measured using correlation coefficient (denoted by R), root mean square error (RMSE), average absolute error (AAE), and max absolute error (MAE). Their definitions are expressed as:

Root mean square error:

$$RMSE = \sqrt{\sum_{i=1}^{n_{error}} \left(y_i - \widehat{y}\right)^2 / n_{error}}$$

Average absolute error:

$$AAE = \sum_{i=1}^{n_{error}} \left|y_i - \widehat{y}\right| / n_{error}$$

Max absolute error:

$$MAE = \max \left|y_i - \widehat{y}\right|, i = 1, \ldots\ldots, n_{error}$$

Table 2 Parameters used in Hartman function with six variables

i	a_{ij}						c_i	p_{ij}					
1	10.0	3.0	17.0	3.5	1.7	8.0	1.0	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.05	10.0	17.0	0.1	8.0	14.0	1.2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	3.0	3.5	1.7	10.0	17.0	8.0	3.0	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650
4	17.0	8.0	0.05	10.0	0.1	14.0	3.2	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

Correlation coefficient:

$$R\left(y,\widehat{y}\right)=\frac{\frac{1}{V}\int\limits_{V}\left(y-\overline{y}\right)\left(\widehat{y}-\overline{\widehat{y}}\right)dv}{\delta(y)\delta\left(\widehat{y}\right)}$$

$$\frac{1}{V}\int\limits_{V}y\widehat{y}dv=\sum\nolimits_{i=1}^{n_{error}}y_i\widehat{y}_i/n_{error},$$

$$\overline{y}=\sum\nolimits_{i=1}^{n_{error}}y_i/n_{error},$$

$$\delta(y)=\sqrt{\sum\nolimits_{i=1}^{n_{error}}\left(y_i-\overline{y}\right)^2/n_{error}}$$

In these four definitions above, n_{error} is the number of the samples in the test set, y_i is the actual response, $\overline{y}$ is average value of actual response, $\widehat{y}$ is the metamodel response, $\overline{\widehat{y}}$ is the average value of metamodel response.

Because the experiments are repeated 1,000 (200 or 500) times, the mean and the coefficient of variation (CV) of R, RMAE, AAE, and MAE are used to evaluate the prediction accuracy of each stand-alone metamodel and ensemble model. The definition of CV is expressed as:

$$CV=\delta/\mu$$

where δ is the standard variance of samples, and μ is the mean of samples.

3.4 Ensemble techniques

There are four techniques considered in this paper: PRS, KRG, SVR, and RBF. These surrogates are used as the four members of the ensemble that is developed based on the several previously described techniques. All the parameters are identified using cross-validation (leave-one-out (LOO) is adopted in this paper) such that they minimize the MSE. In all the above-mentioned surrogates, the following parameters should be identified: the highest order (denoted by d) in PRS, the parameter (c) in multiquadrics of RBF, the parameter (θ) in Gassian correlation function of Kriging, and the parameter (C, ε, σ) in SVR. The LOO cross-validation results are presented in Table 4. The mathematical descriptions of the five metamodels are provided in the Appendix A.

4 Results and analysis of experiments

Part of the marks used to label the ensemble techniques is inherited from Acar and Rais-Rohani (2009). The model based on the simple average is denoted by EA; the one based on the heuristic method of Goel et al. (2007) is labeled as EG; the one based on the prediction variance of Zerpa et al. (2005) is denoted by EV; the one based on minimizing PRESS (GMSE) in Acar and Rais-Rohani (2009) is labeled as EP; the one based on minimizing RMSE^v in Acar and Rais-Rohani (2009) is labeled as EM; OWS_{diag} in Viana et al. (2009) is denoted by Od; BestPRESS is denoted by BP; and the one proposed in this paper is denoted by ER. The results of different benchmark problems are shown with the help of boxplots (the description of boxplot is provided in the Appendix B), and the means and CVs of the error metrics are presented with several tables. Additionally, to facilitate comparison of the performances of the the ensembles and single surrogates, the frequencies of the rank of them in terms of R, RMSE, AAE, and MAE are also presented with other several tables.

4.1 Correlation coefficient

The correlation coefficients for different test functions are shown in Fig. 1, from which we can see: (1) No single metamodel works best for all test functions and correlation coefficient for different stand-alone metamodel varied with DOE significantly; the eight ensemble models work better than the worst stand-alone metamodel, and correlation coefficient for ensemble model varied with DOE insignificantly; (2) In almost all of the test problems, although EM and ER have similar median, and have better performance than the other ensemble models, EM has longer tail, which indicates that EM is less robust than ER; (3) EP has the worst performance among all the ensembles for A, B, and C; (4) BP has the second worst performance in A and B, and has the worst performance in D, which reveals that BP can not capture the real error perfectly, that is, BP can not find the best single surrogate according to the cross-validation in most of the replications; and (5) At last, it is worthy noting that, in all the test problems, EG and Od have the similar results.

Table 4 Summary of LOO cross-validation results for the parameters in all of the surrogates

Benchmark problem	d in PRS	θ in KRG	C, ε, σ in SVR	c in RBF
Branin–Hoo	2	7	$C=1e3, \varepsilon=0.15, \sigma=7$	5
Camelback	2	12	$C=1e3, \varepsilon=0.15, \sigma=7$	0
Goldstein–Price	2	12	$C=1e6, \varepsilon=1e-4, \sigma=1$	2
Hartman-3	2	8	$C=10, \varepsilon=0.015, \sigma=0.7$	2
Hartman-6	4	8	$C=10, \varepsilon=0.015, \sigma=0.7$	2
Abalone	2	40	$C=1e3, \varepsilon=0.01, \sigma=4$	0

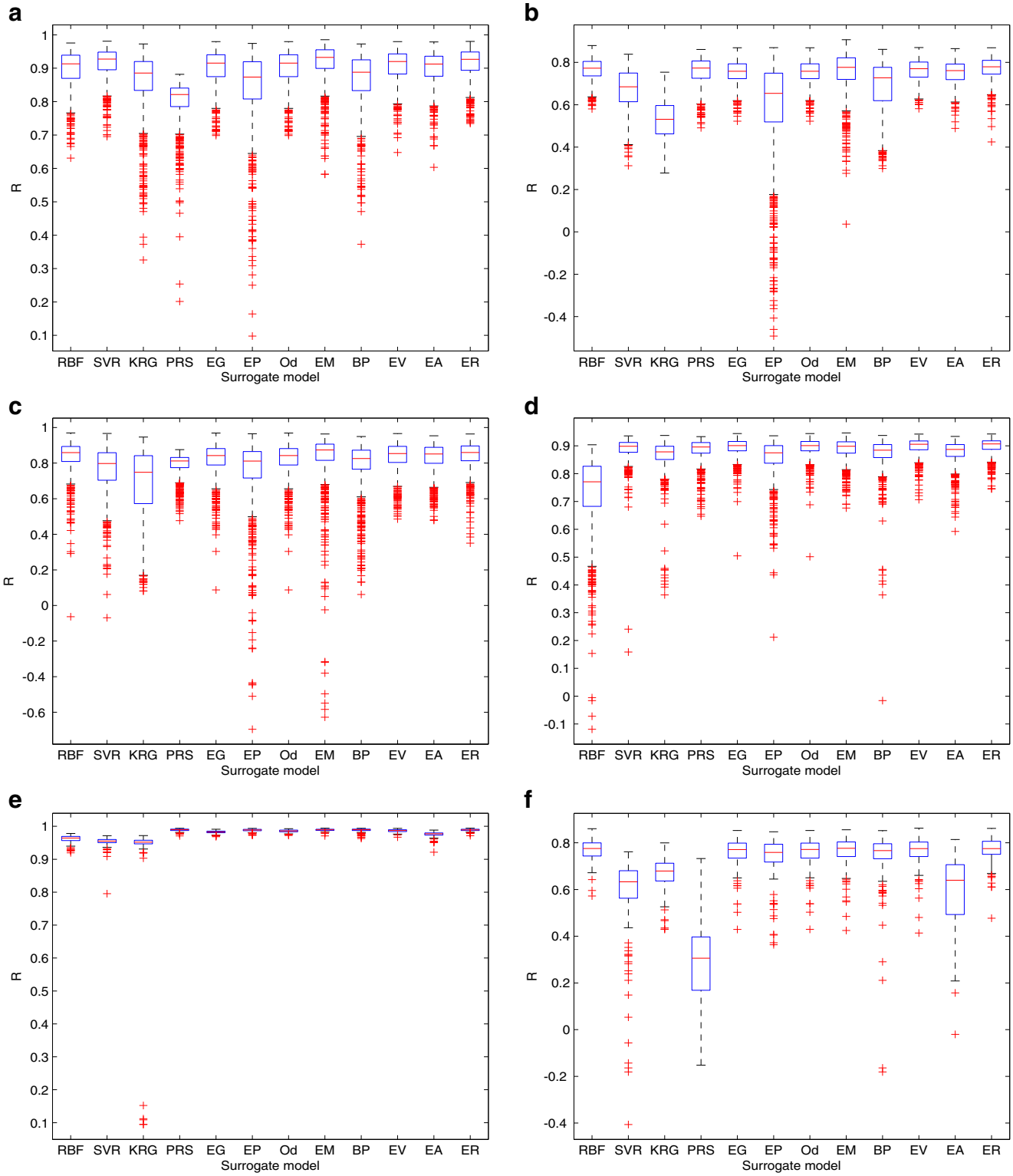

Fig. 1 Correlations between actual and predicted response for different surrogate models. **a** Branin–Hoo, **b** Camelback, **c** Goldstein–Price, **d** Hartman-3, **e** Hartman-6, **f** Abalone

Table 5 Mean and CV (in parenthesis) of correlation coefficient between actual and predicted response (based on 1,000/200/500 DOEs) for different metamodels, the highest value in each category is shown in bold for ease of comparison

	Branin–Hoo	Camelback	Goldstein–Price	Hartman-3	Hartman-6	Abalone
RBF	0.8985 (0.0606)	0.7678 (0.065)	0.8386 (0.1042)	0.7351 (0.1806)	0.962 (0.011)	0.7703 (0.0573)
SVR	0.9155 (0.0495)	0.6729 (0.1427)	0.7661 (0.1708)	0.889 (0.0497)	0.9532 (0.0153)	0.5866 (0.2908)
KRG	0.8616 (0.1041)	0.5276 (0.1713)	0.6826 (0.3009)	0.8672 (0.0666)	0.9303 (0.1416)	0.6681 (0.0985)
PRS	0.8017 (0.0816)	0.7583 (0.0829)	0.7941 (0.0746)	0.8862 (0.0433)	0.989 (0.0036)	0.2798 (0.5704)
EG	0.9025 (0.0559)	0.754 (0.0728)	0.8224 (0.1118)	0.8943 (0.0364)	0.9826 (0.004)	0.7580 (0.0792)
EP	0.845 (0.1332)	0.5919 (0.3809)	0.7742 (0.2109)	0.8589 (0.0771)	0.9882 (0.0037)	0.7428 (0.1093)
Od	0.9025 (0.0559)	0.754 (0.0728)	0.8224 (0.1118)	0.8944 (0.0365)	0.9856 (0.0037)	0.7581 (0.0790)
EM	**0.9175 (0.0608)**	0.7549 (0.1305)	0.8302 (0.1951)	0.889 (0.0427)	**0.9891 (0.0031)**	0.7638 (0.0821)
BP	0.8689 (0.0913)	0.6862 (0.1798)	0.7904 (0.1695)	0.8725 (0.0687)	0.9883 (0.0051)	0.7404 (0.1654)
EV	0.9084 (0.0518)	0.7642 (0.0672)	0.8369 (0.0951)	0.8978 (0.0332)	0.9865 (0.004)	0.7648 (0.0783)
EA	0.9006 (0.0538)	0.7538 (0.071)	0.8329 (0.0969)	0.8767 (0.0483)	0.9761 (0.0072)	0.5846 (0.2694)
ER	0.9153 (0.049)	**0.7724 (0.0682)**	**0.8434 (0.0911)**	**0.9002 (0.0298)**	0.9888 (0.0033)	**0.7717 (0.0663)**

Table 5 shows the mean and the coefficient of variation for different test functions to assess the performance of different metamodels. It is clear that the average correlation coefficient for ER was the best for almost all the test functions except Branin–Hoo and Hartman-6. On the contrary, EM has a best performance in Branin–Hoo and Hartman-6. In addition, it is interesting that, in low dimensional problems, such as Branin–Hoo, Camelback, and Goldstein–Price, EG and Od have exactly the same result, and in high dimensional problems, such as Hartman-3, Hartman-6, and Abalone, although their results are not the same, their results are similar. Combining Table 6 to Table 5, we can find that besides four times of 1st, there are two times of 3rd in ER, that is, ER has an ideal result in all of the six test problems, which indicates ER has a robust prediction capacity. On the other hand, the performances of the other ensembles and all the individual surrogates vary apparently with test problems. Even the second best ensemble, EM, performs well just in two test problems, but in the other problems, it doesn't perform perfectly, just one time of 4th and three times of 5th. The third best model is RBF. Though RBF is inferior to ER and EM, it is still the best in all of the individual surrogates, and it seems has certain reasonable robust results.

4.2 RMSE

Next, we compare different metamodels based on the RMSE in predictions at test points. As shown in Fig. 2, we can see: (1) RBF has the best performance in all of the stand-alone metamodels in problem B, C, and F, its prediction accurate is par with the best ensemble model; In addition, PRS was either the best or the second best for all the test problems in all of the stand-alone metamodels; (2) Generally, all of these eight ensemble models are better than the worst stand-alone metamodel, and RMSE for ensemble models didn't vary with DOE significantly,

Table 6 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for all the benchmark problems and Abalone problem (the total number of problems is six), and the error metric is correlation coefficient

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	–	3	–	–	–	–	–	1	–	1	–	1	6
SVR	–	1	–	–	–	1	–	–	–	2	2	–	6
KRG	–	–	–	–	–	–	–	–	1	2	–	3	6
PRS	–	1	–	1	–	–	1	1	–	–	–	2	6
EG	–	–	–	1	1	2	1	1	–	–	–	–	6
EP	–	–	–	–	1	–	1	–	–	1	3	–	6
Od	–	–	1	–	1	2	2	–	–	–	–	–	6
EM	2	–	–	1	3	–	–	–	–	–	–	–	6
BP	–	–	–	1	–	–	–	1	4	–	–	–	6
EV	–	1	3	1		1	–	–	–	–	–	–	6
EA	–	–	–	1	–	–	1	2	1	–	1	–	6
ER	4	–	2	–	–	–	–	–	–	–	–	–	6

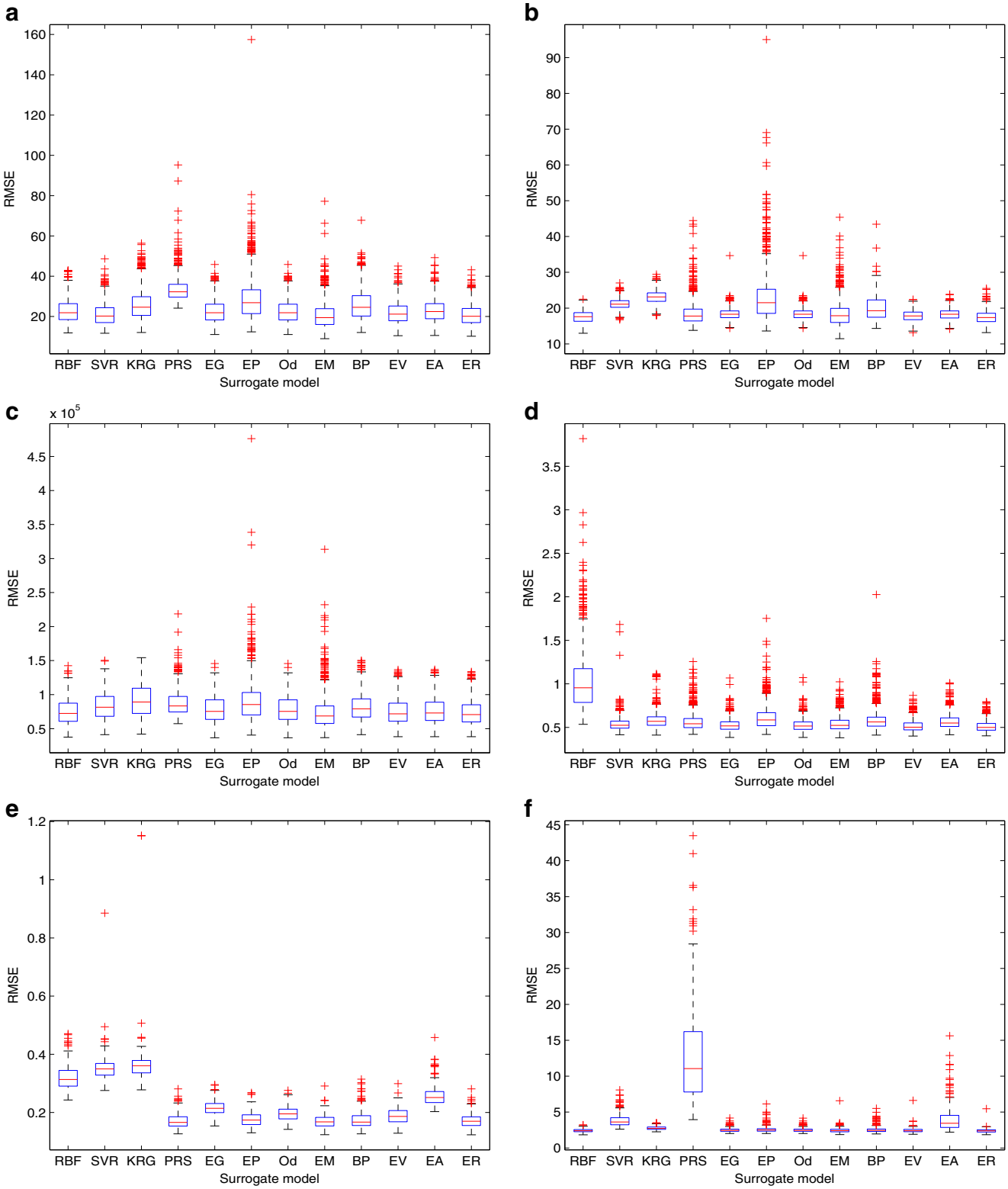

Fig. 2 RMSE for different surrogate models. **a** Branin–Hoo, **b** Camelback, **c** Goldstein–Price, **d** Hartman-3, **e** Hartman-6, **f** Abalone

Table 7 Mean and CV of RMSE for different metamodels, the lowest value in each category is shown in bold for ease of comparison

	Branin–Hoo	Camelback	Goldstein–Price	Hartman-3	Hartman-6	Abalone
RBF	22.7999 (0.2556)	17.5359 (0.0953)	75,561.4431 (0.2435)	1.03 (0.335)	0.3205 (0.136)	2.4271 (0.0937)
SVR	21.0289 (0.2527)	21.1345 (0.07)	83,727.7761 (0.2431)	0.5388 (0.1577)	0.353 (0.1438)	3.8816 (0.2447)
KRG	26.0097 (0.2833)	23.0588 (0.0748)	91,279.917 (0.2529)	0.5824 (0.1552)	0.3798 (0.3383)	2.7764 (0.0848)
PRS	33.5966 (0.184)	18.5474 (0.1812)	87,716.7881 (0.2054)	0.5642 (0.1834)	0.1716 (0.1526)	13.1735 (0.5685)
EG	22.6632 (0.2488)	18.3123 (0.0866)	78,633.1366 (0.2502)	0.5275 (0.1351)	0.2166 (0.112)	2.5157 (0.1227)
EP	28.6822 (0.3768)	23.127 (0.3099)	86,662.5008 (0.3136)	0.6132 (0.2253)	0.1772 (0.1471)	2.6141 (0.2013)
Od	22.6632 (0.2488)	18.3123 (0.0866)	78,633.1366 (0.2502)	0.5275 (0.1363)	0.1965 (0.1278)	2.5152 (0.1224)
EM	**20.7718 (0.3358)**	18.495 (0.2002)	74,107.4594 (0.3402)	0.5417 (0.1602)	**0.1697 (0.1354)**	2.4633 (0.1664)
BP	25.7626 (0.2828)	19.8679 (0.1606)	81,685.6578 (0.2426)	0.5858 (0.2042)	0.1758 (0.1893)	2.5812 (0.2004)
EV	21.8817 (0.243)	17.8344 (0.0853)	75,288.8687 (0.2555)	0.5167 (0.1288)	0.1892 (0.1419)	2.4617 (0.1625)
EA	23.0023 (0.2299)	18.2824 (0.0795)	76,377.3937 (0.2541)	0.5709 (0.1502)	0.2579 (0.144)	4.1356 (0.4929)
ER	20.9243 (0.2487)	**17.4485 (0.0981)**	**73,666.9996 (0.248)**	**0.5112 (0.1216)**	0.1722 (0.1398)	**2.4053 (0.1316)**

which suggestes the ensemble models are more robust; (3) Stand-alone model on the whole has worse prediction accuracy than ensemble model, which indicates the necessity of adopting the ensemble techniques; and (4) The technique of ER proposed in this paper has better performance than the other ensemble models in RMSE.

Table 7 shows that the average RMSE for ER was the best for almost all the test functions except Branin-Hoo and Hartman-6. Although the average RMSE for ER in Branin–Hoo is gently larger than EM, ER has a lower CV, which indicates that ER is more robust than EM in Branin–Hoo.

Table 8 complements Table 7 and shows the frequencies of the rank of all the ensembles and individual surrogates in the ensembles. From the table, we can see that the result is similar to that in Table 6. For all of the benchmark problems and Abalone problem, ER is the first for four times, is the second for one time, and is the third for one time. Apparently, ER is the best model in all of the ensembles and individual models in terms of RMSE. The second best model is EM, and the third best model is the single surrogate RBF.

4.3 AAE

Figure 3 shows the AAE for different metamodels on different test functions. It shows us the following findings: (1) For problem A, PRS has a higher AAE than the rest individual metamodels; there are three individual metamodels which have similar AAEs, which may be the reason why these eight ensemble models also have similar AAEs. (2) For the test problem D, E, and F, the ensemble models have significantly lower AAE than the worst individual surrogate. (3) For problem F, PRS has the worst result, which possibly suggest that PRS is actually not suitable for such kind of problems; and because of PRS's bad performance, EA has a similarly bad result. (4) Being similar to PRS in F, RBF is also not suit for D; but in A, B, C and F, RBF has ideal results, which indicates the performance of surrogate

Table 8 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for all the benchmark problems and Abalone problem (the total number of problems is six), and the error metric is RMSE

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	–	2	–	1	–	–	1	–	–	1	–	1	6
SVR	–	–	1	–	1	–	–	–	1	2	1	–	6
KRG	–	–	–	–	–	–	–	–	2	1	1	2	6
PRS	–	1	–	–	–	–	1	1	–	–	1	2	6
EG	–	–	1	–	1	2	1	1	–	–	–	–	6
EP	–	–	–	–	1	–	–	1	–	1	2	1	6
Od	–	–	–	1	2	2	1	–	–	–	–	–	6
EM	2	1	–	1	–	1	1	–	–	–	–	–	6
BP	–	–	–	1	–	–	1	1	2	1	–	–	6
EV	–	1	3	1	–	1	–	–	–	–	–	–	6
EA	–	–	–	1	1	–	–	2	1	–	1	–	6
ER	4	1	1	–	–	–	–	–	–	–	–	–	6

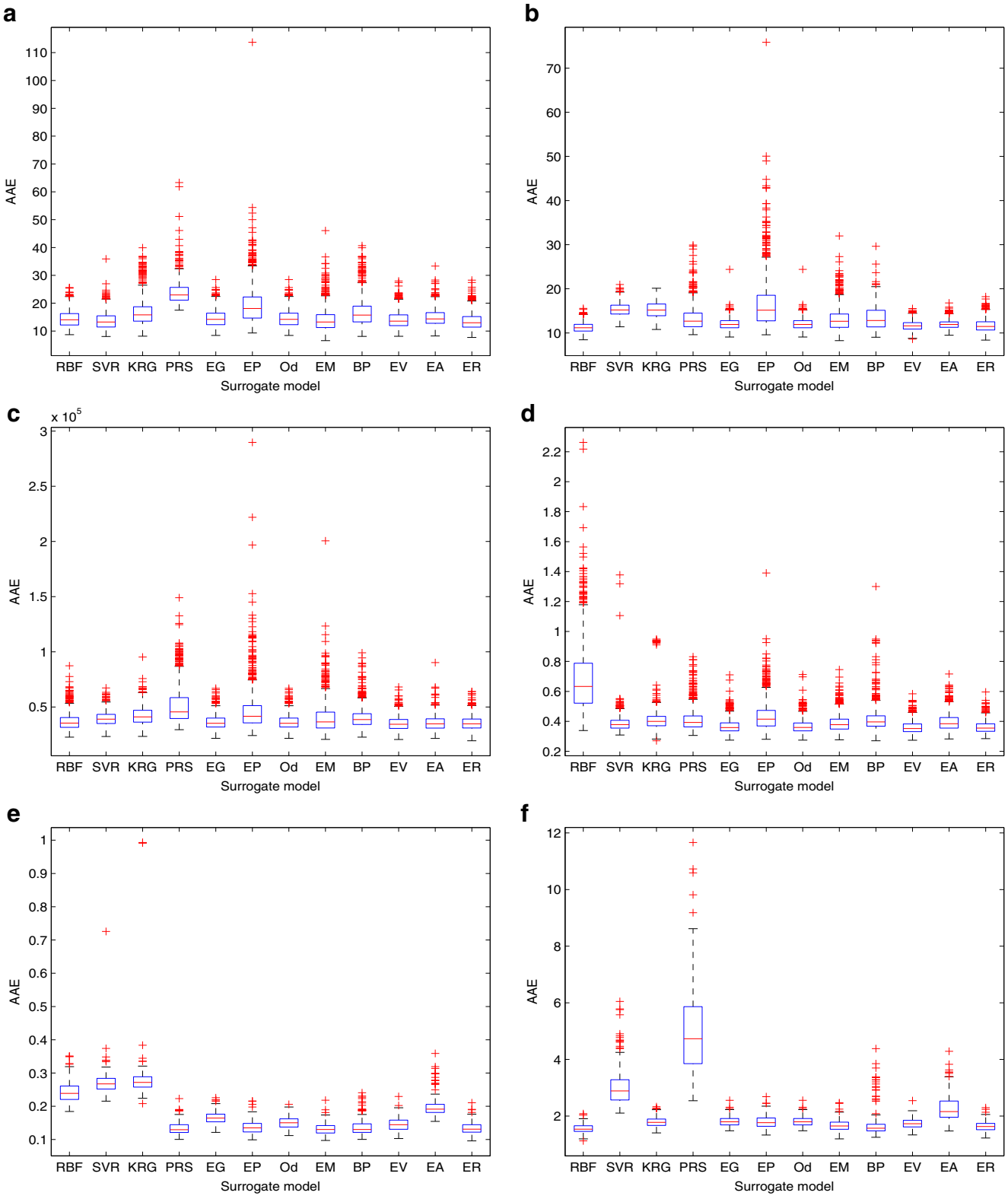

Fig. 3 AAE for different surrogate models. **a** Branin–Hoo, **b** Camelback, **c** Goldstein–Price, **d** Hartman-3, **e** Hartman-6, **f** Abalone

Table 9 Mean and CV of AAE for different metamodels, the lowest value in each category is shown in bold for ease of comparison

	Branin–Hoo	Camelback	Goldstein–Price	Hartman-3	Hartman-6	Abalone
RBF	14.4519 (0.2082)	**11.2727 (0.1027)**	36,679.175 (0.2101)	0.6827 (0.3337)	0.2431 (0.1293)	**1.556 (0.1014)**
SVR	13.73 (0.2219)	15.2986 (0.0949)	39,375.2395 (0.1612)	0.3877 (0.1635)	0.2709 (0.1491)	3.0463 (0.2319)
KRG	16.7237 (0.2812)	15.245 (0.1117)	41,927.5562 (0.1906)	0.4087 (0.1751)	0.2911 (0.3955)	1.7918 (0.0981)
PRS	23.7886 (0.1749)	13.2713 (0.1991)	51,316.771 (0.3272)	0.4096 (0.1776)	0.1343 (0.1456)	5.0286 (0.3196)
EG	14.5383 (0.2088)	12.0577 (0.1009)	36,098.5254 (0.1814)	0.367 (0.1284)	0.1658 (0.1078)	1.8129 (0.0953)
EP	19.3623 (0.373)	16.542 (0.3475)	43,921.8599 (0.3823)	0.4354 (0.2226)	0.1374 (0.1416)	1.7974 (0.1275)
Od	14.5383 (0.2088)	12.0577 (0.1009)	36,098.5254 (0.1814)	0.3672 (0.129)	0.151 (0.1231)	1.8125 (0.0953)
EM	14.0549 (0.3019)	13.1827 (0.2077)	40,091.8514 (0.3533)	0.3881 (0.153)	**0.1322 (0.1269)**	1.6653 (0.1198)
BP	16.6824 (0.2864)	13.374 (0.1902)	40,112.9957 (0.2347)	0.4127 (0.1961)	0.1374 (0.1795)	1.7165 (0.2976)
EV	13.995 (0.2122)	11.6794 (0.091)	**35,012.6802 (0.1796)**	**0.361 (0.1166)**	0.1457 (0.1346)	1.7449 (0.1063)
EA	14.8322 (0.2016)	11.9761 (0.0824)	35,495.4535 (0.1881)	0.3955 (0.1468)	0.1984 (0.1531)	2.2763 (0.1933)
ER	**13.5391 (0.2212)**	11.7025 (0.1216)	35,340.7912 (0.1844)	0.3615 (0.1096)	0.1339 (0.1303)	1.6335 (0.1103)

is problem-dependent. Additionally, from Table 9, we can see that RBF performance best in Camelback and Abalone, EV performance best in Goldstein–Price and Hartman-6, and ER performance best just in Branin-Hoo. In addition, Table 10 shows that RBF and EV have the highest frequency of 1st, and the EM and ER have the second highest frequency of 1st. But ER has the highest times (four times) of 2nd in all of the ensembles and individual surrogates. Combining the times in the 1st, 2nd, and 3rd, and considering the robustness, we think that the best robust model should be ER, the second should be EV, and the third should be EV.

4.4 MAE

Next, the MAEs of different metamodels for different test functions are compared. Figure 4 shows that, for A and C, all of these models, including ensembles and single surrogates, have similar MAEs, but in the other problems, the difference in MAE is apparent; for B, EP has a worst performance in all of the ensembles, and it has a long tail in the figure, which means it has a larger deviation; for D and F, the worst models is RBF and PRS respectively.

Numerical quantification of the results is given in Table 11, where we can observe that ER is not the best model in all of the problems, EM perform best in three test problems, and other three single surrogates all perform best in one problem. With the help of Table 12, we also find that the best model may be EM. But combining the times of the 1st, 2nd, and 3rd, it is easy to find that ER is also a more reasonable robust model than the single surrogate, such as, RBF, SVR, and PRS.

4.5 The effect of the number of the validation points

All of the results above all are under the consideration of $V = 0.8N$ in ER, EM, Od, and EV. In order to examine the effect of the number of the validation points V on the prediction results of all the ensemble surrogates, $V = 0.3N$ and $V = 0.5N$ are also considered in the following experiments. Considering the length of this article, however, we just

Table 10 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for all the benchmark problems and Abalone problem (the total number of problems is six), and the error metric is AAE

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	2	–	–	–	1	1	–	–	–	1	–	1	6
SVR	–	1	–	–	1	–	1	–	–	–	3	–	6
KRG	–	–	–	–	–	1	–	1	–	3	–	1	6
PRS	–	–	1	–	–	–	–	1	1	–	–	3	6
EG	–	–	1	–	2	1	–	1	1	–	–	–	6
EP	–	–	–	–	1	–	1	–	–	–	3	1	6
Od	–	–	–	2	–	1	2	1	–	–	–	–	6
EM	1	–	1	1	–	1	1	1	–	–	–	–	6
BP	–	–	–	2	–	–	–	–	3	1	–	–	6
EV	2	1	1	–	1	1	–	–	–	–	–	–	6
EA	–	–	1	1	–	–	1	1	1	1	–	–	6
ER	1	4	1	–	–	–	–	–	–	–	–	–	6

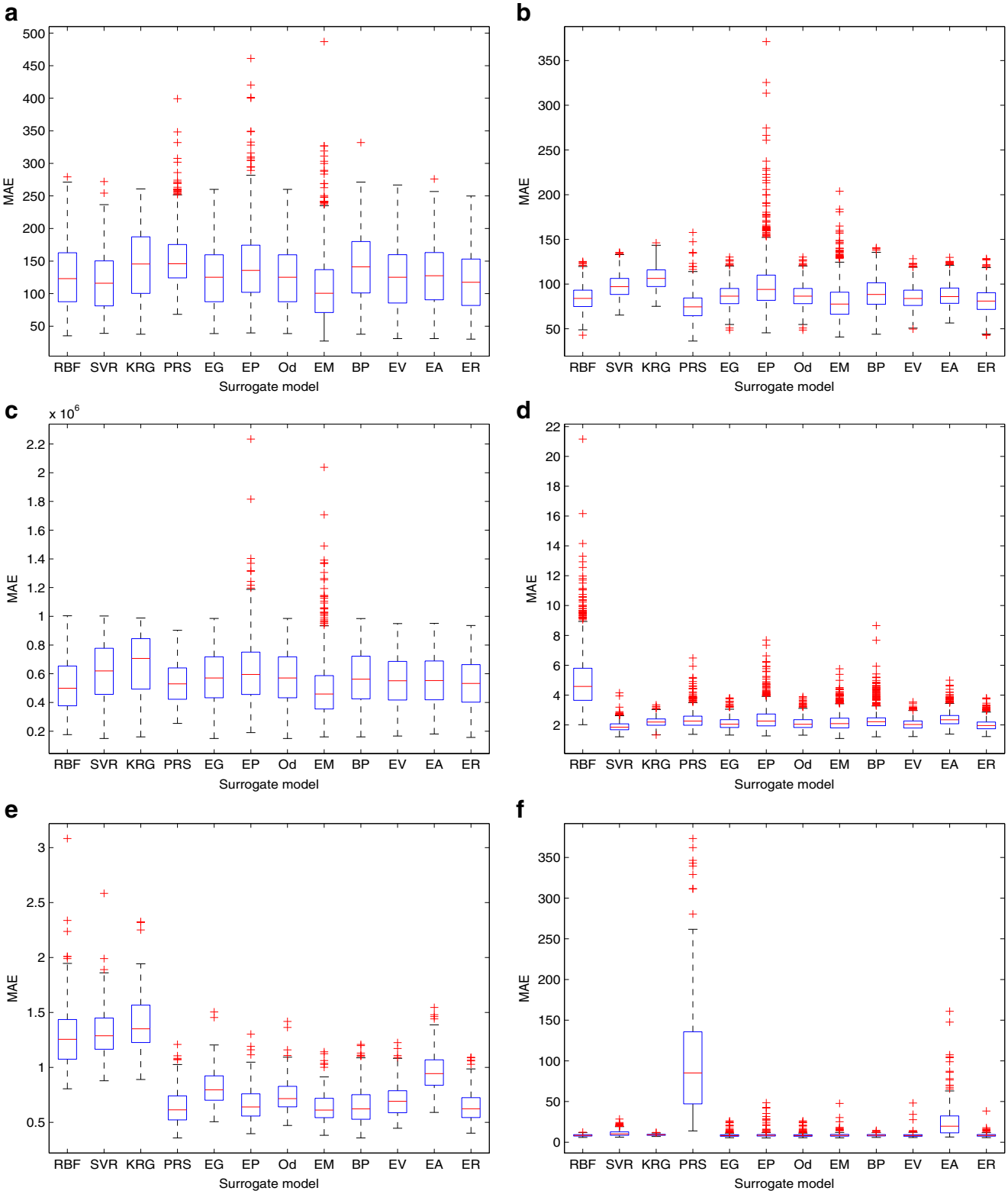

Fig. 4 MAE for different surrogate models. **a** Branin–Hoo, **b** Camelback, **c** Goldstein–Price, **d** Hartman-3, **e** Hartman-6, **f** Abalone

Table 11 Mean and CV of MAE for different metamodels, the lowest value in each category is shown in bold for ease of comparison

	Branin–Hoo	Camelback	Goldstein–Price	Hartman-3	Hartman-6	Abalone
RBF	127.6625 (0.3819)	84.4234 (0.1566)	518,555.0576 (0.3305)	5.0071 (0.3943)	1.2935 (0.2311)	**8.4792 (0.1422)**
SVR	118.6682 (0.3763)	97.7798 (0.1332)	616,152.7969 (0.3028)	**1.8825 (0.1665)**	1.3168 (0.1725)	11.328 (0.3425)
KRG	144.5638 (0.3585)	106.9659 (0.1243)	667,815.863 (0.3049)	2.1898 (0.1447)	1.4146 (0.1877)	9.235 (0.0879)
PRS	152.6863 (0.2804)	**75.1959 (0.2063)**	537,426.472 (0.2663)	2.3453 (0.2376)	0.6422 (0.2529)	104.9109 (0.7085)
EG	126.265 (0.3681)	86.9356 (0.1472)	572,094.3508 (0.3101)	2.1057 (0.19)	0.8178 (0.1947)	8.9009 (0.3601)
EP	142.6643 (0.3916)	100.0874 (0.3107)	592,091.0514 (0.3333)	2.4378 (0.3166)	0.672 (0.2357)	9.8579 (0.6017)
Od	126.2649 (0.3681)	86.9355 (0.1472)	572,094.3508 (0.3101)	2.1027 (0.1921)	0.7385 (0.2052)	8.8962 (0.359)
EM	**109.7812 (0.4649)**	80.4475 (0.2613)	**495,956.4763 (0.4159)**	2.1948 (0.2682)	**0.6392 (0.213)**	9.1818 (0.4288)
BP	141.3762 (0.3511)	89.953 (0.1961)	573,893.8504 (0.3195)	2.3108 (0.2833)	0.6588 (0.2743)	8.7025 (0.1599)
EV	125.4616 (0.3697)	85.3022 (0.1538)	554,610.7838 (0.3053)	2.0448 (0.1763)	0.7118 (0.2158)	8.8935 (0.4399)
EA	128.5255 (0.3544)	87.7427 (0.1471)	558,522.7134 (0.303)	2.3961 (0.2009)	0.9642 (0.1889)	27.0526 (0.8824)
ER	120.3731 (0.3747)	81.6007 (0.1766)	536,010.0076 (0.3106)	1.9944 (0.1839)	0.6435 (0.2171)	8.7271 (0.3136)

take *Camelback* as an example. Different from Tables 5–12, where the Rs (or RMSEs; AAEs; MAEs) of all the test problems are get together in a same table, here, we get together the R, RMSE, AAE, and MAE for *Camelback* and presented them in a same table, thereby, the total number is four (the number of the error metrics (R, RMSE, AAE, and MAE)) rather than six (the number of the test problems). Tables 13, 14 and 15 presents the results for $V = 0.3N$, $V = 0.5N$, and $V = 0.8N$ respectively. From the three tables, we can obtain the following findings: (1) The prediction accuracies of the ensemble models (ER, EM, Od, EV), which base on the validation points, improved with the increasing number of validation points; (2) Nevertheless, their speed of improvement is different; varying from $V = 0.3N$ to $V = 0.8N$, ER has an apparent improvement, the frequency of 1st improves from zero to two. on the other hand, the improvement in EM is not so apparent; and (3) when $V = 0.3N$, RBF has the best performance, so, when the validation points is not easy to obtain, choosing a single surrogate may be a reasonable strategy, but in practice, we have no the prior knowledge about which is the best single surrogate.

Additionally, we should point out that the performance of BP (BestPRESS) is not ideal according to the results presented in Tables 5–15, which may suggest (1) it is difficult for cross-validation to capture the real errors, so, the best single surrogate can not be picked out according to the cross-validation; and (2) even if it can perfectly estimate the real error, its prediction accuracy would only be similar to the best single surrogate (after all, BestPRESS is like assigning a unit weight for the surrogate with smallest PRESS and zeroing all the others), but according to the experiments results, the capacity of single surrogate may be worse than ensemble models. Finally, we compare the efficiency between EM and ER, because they are both based on minimizing RSME (or prediction MSE). The time consumption of EM and ER is presented in Table 16. In this experience, we choose a low dimensional problem, a median high dimensional problem, and a high dimensional problem as test problems. From the table, we can see that in low

Table 12 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for all the benchmark problems and Abalone problem (the total number of problems is six), and the error metric is MAE

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	1	1	–	1	–	–	1	–	–	1	–	1	6
SVR	1	1	–	–	–	–	–	–	–	2	2	–	6
KRG	–	–	–	–	–	1	–	1	–	–	1	3	6
PRS	1	1	–	1	–	–	–	–	1	–	–	2	6
EG	–	–	–	–	1	2	1	2	–	–	–	–	6
EP	–	–	–	–	1	–	–	–	1	2	2	–	6
Od	–	–	–	1	2	1	2	–	–	–	–	–	6
EM	3	1	–	–	–	–	2	–	–	–	–	–	6
BP	–	1	–	1	–	–	–	1	3	–	–	–	6
EV	–	–	1	2	2	1	–	–	–	–	–	–	6
EA	–	–	–	–	–	1	–	2	1	1	1	–	6
ER	–	1	5	–	–	–	–	–	–	–	–	–	6

Table 13 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for Camelback in terms of the error metrics: R, RMSE, AAE, and MAE; the validation point $V = 0.3N$

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	3	–	1	–	–	–	–	–	–	–	–	–	4
SVR	–	–	–	–	–	–	–	–	4	–	–	–	4
KRG	–	–	–	–	–	–	–	–	–	2	1	1	4
PRS	1	1	–	–	–	–	2	–	–	–	–	–	4
EG	–	–	–	2	1	1	–	–	–	–	–	–	4
EP	–	–	–	–	–	–	–	–	–	2	2	–	4
Od	–	–	–	1	3	–	–	–	–	–	–	–	4
EM	–	–	–	–	–	–	–	–	–	–	1	3	4
BP	–	–	–	–	–	–	–	4	–	–	–	–	4
EV	–	1	2	1	–	–	–	–	–	–	–	–	4
EA	–	–	1	–	–	2	1	–	–	–	–	–	4
ER	–	2	–	–	–	1	1	–	–	–	–	–	4

Table 14 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for Camelback in terms of the error metrics: R, RMSE, AAE, and MAE; the validation point $V = 0.5N$

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	1	1	1	1	–	–	–	–	–	–	–	–	4
SVR	–	–	–	–	–	–	–	–	–	–	3	1	4
KRG	–	–	–	–	–	–	–	–	–	–	1	3	4
PRS	1	–	–	–	–	–	–	1	1	1	–	–	4
EG	–	–	–	–	2	1	1	–	–	–	–	–	4
EP	–	–	–	–	–	–	–	2	2	–	–	–	4
Od	–	–	–	–	1	3	–	–	–	–	–	–	4
EM	1	–	2	1	–	–	–	–	–	–	–	–	4
BP	–	–	–	–	–	–	–	1	–	3	–	–	4
EV	–	–	1	2	1	–	–	–	–	–	–	–	4
EA	–	–	–	–	–	–	3	–	1	–	–	–	4
ER	1	3	–	–	–	–	–	–	–	–	–	–	4

Table 15 Frequency of the rank of the ensemble surrogates and the individual surrogates in the ensembles for Camelback in terms of the error metrics: R, RMSE, AAE, and MAE; the validation point $V = 0.8N$

	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	Total
RBF	1	2	–	1	–	–	–	–	–	–	–	–	4
SVR	–	–	–	–	–	–	–	–	–	3	1	–	4
KRG	–	–	–	–	–	–	–	–	–	1	1	2	4
PRS	1	–	–	1	–	–	–	2	–	–	–	–	4
EG	–	–	–	–	1	2	1	–	–	–	–	–	4
EP	–	–	–	–	–	–	–	–	–	–	2	2	4
Od	–	–	–	–	1	2	1	–	–	–	–	–	4
EM	–	1	–	–	1	–	2	–	–	–	–	–	4
BP	–	–	–	–	–	–	–	–	4	–	–	–	4
EV	–	1	2	–	1	–	–	–	–	–	–	–	4
EA	–	–	–	2	–	–	–	2	–	–	–	–	4
ER	2	–	2	–	–	–	–	–	–	–	–	–	4

Table 16 Comparison of the time cost between the processes of obtaining the weights in ER and EP (1,000 times replications in BH, and 200 times replications in Hartman-6; run time in EP is denoted by EP_{tim}, and run time in ER is denoted by ER_{tim}), the run times are the mean values

Benchmark problems	EP_{tim}	ER_{tim}	EP_{tim}/ER_{tim} ratio	Number of iterations in ER	*tol* in ER
BH	0.0137	0.0215	1.57	47.773	0.1
Hartman3	0.0016	0.0227	14.19	8.513	0.1
Hartman6	0.0026	0.0307	7.935	11.81	0.1

dimensional problem BH (two dimensions), the cost in time consumption using EP is nearly two times as much as that in ER; for median high dimensional problem Hartman-3 (three dimensions), the cost of time consumption using EP is 14.19 times as much as that in ER; Furthermore, in high dimensional problem Hartman-6 (six dimensions), the cost in time consumption using EP is 7.935 times as much as that in ER. The experiment result reveals that when the dimension in problem is large (especially when dozens of variables appear in real-life problems), choosing recursive arithmetic average ensemble technique rather than the ensemble techniques based on optimization process may be a reasonable strategy. The results support the viewpoint presented in the last paragraph of Section 2.3.2.

5 Conclusion

In this paper, we examined several existing combining techniques, proposed recursive arithmetic average ensemble technique, and finally discussed the experiment results.

1. After examination of the existing combining techniques, we find (1) OWS_{idea} is essentially the same as EM; and (2) OWS is also the same as EP. The difference between them is just the expression used to obtain the weights.
2. After examination of the results for these five test functions and Abalone problem, we can see clearly that the ensemble technique proposed in this paper has more significant prediction accuracy than stand-alone metamodels in most problems, and for almost all of problems presented in this paper even surpasses the previously reported ensemble techniques.
3. Because of adopting cross validation in choosing of the best parameters in stand-alone metamodels, all of the models, including individual models and ensemble models, have significantly improved their prediction accuracy.
4. EG and Od have the similar results in terms of R, RMSE, AAE, MAE in all of the test problems, especially in low dimensional problem. The cause is that EG and Od have the similar structure, which we have pointed out in Section 1.
5. In this paper, we limit our conclusion to low dimension problems (less than seven dimensions), what about the high dimension problems is our future research work.

Although the technique proposed in this paper achieves desirable results, the advantages of combination over selection are still difficult to clarify (Yang 2003). This is, despite our efforts, we are still operating using the "insurance policy" mode rather than offering substantial improvements. In addition, finding more efficient methods to improve the prediction accuracy of the ensemble model is also our future work.

Acknowledgments The funding provided for this study by the National Science Foundation of China under Grant NO.70931002 and NO.70672088 is gratefully acknowledged.

Appendix A: Several metamodeling techniques

Here, there are four metamodeling techniques (PRS, RBF, Kriging, SVR) are considered.

A.1 PRS

For PRS, the highest order is allowed to be 4 in this paper, but the used order in a specific problem is determined by the selected sample set. When the highest order of a polynomial model is 4, it can be expressed as:

$$\widetilde{F}(x) = a_0 + \sum_{i=1}^{N} b_i x_i + \sum_{i=1}^{N} c_{ii} x_{i_i}^2 + \sum_{ij(i<j)}^{c_{ij}x_i x_j} + \sum_{i=1}^{N} d_i x_{i_i}^3 + \sum_{i=1}^{N} e_i x_{i_i}^4 \tag{21}$$

where $\widetilde{F}$ is the response surface approximation of the actual response function, N is the number of variables in the input vector $\mathbf{x}$, and a, b, c, d, e are the unknown coefficients to be determined by the least squares technique.

Notice that 3rd and 4th order models in polynomial model do not have any mixed polynomial terms (interactions) of order 3 and 4. Only pure cubic and quadratic terms are included to reduce the amount of data required for model

construction. A lower order model (Linear, Quadratic, and Cubic) includes only lower order polynomial terms (only linear, quadratic, or cubic terms correspondingly).

A.2 RBF

The general form of the RBF approximation can be expressed as:

$$f(x) = \sum_{i=1}^{m} \beta_i \varphi\left(\left\| x - x_i \right\|\right) \tag{22}$$

Powell (1987) considers several forms for the basis function $\varphi(\cdot)$:

1. $\varphi(r) = e^{\left(-r^2/c^2\right)}$ Gaussian
2. $\varphi(r) = (r^2 + c^2)^{\frac{1}{2}}$ Multiquadrics
3. $\varphi(r) = (r^2 + c^2)^{-\frac{1}{2}}$ Reciprocal Multiquadrics
4. $\varphi(r) = \left(r/c^2\right) \log\left(r/c\right)$ Thin-Plate Spline
5. $\varphi(r) = \frac{1}{1+e^{r/c}}$ Logistic

where $c \geq 0$. Particularly, the multi-quadratic RBF form has been applied by Meckesheimer et al. (2001, 2002) to construct an approximation after Hardy (1971), who used linear combinations of a radically symmetric function based on the Euclidean distance of the form:

$$\varphi(x) = \beta_0 + \sum_{i=1}^{n} \beta \left\| \mathbf{x} - \mathbf{x}_i \right\| \tag{23}$$

where $\|\cdot\|$ represents the Euclidean norm. Replacing $\varphi(x)$ with the vector of response observations, $\mathbf{y}$ yields a linear system of n equations and n variables, which is used to solve β. As described above, this technique can be viewed as an interpolating process. RBF surrogates have produced good fits to arbitrary contours of both deterministic and stochastic responses (Powell 1987). Different RBF forms were compared by McDonald et al. (2000) on a hydro code simulation, and the author found that the Gaussian and the multi-quadratic RBF forms performed best generally.

A.3 Kriging

For computer experiments, kriging is viewed from a Bayesian perspective where the response is regarded as a realization of a stationary random process. The general form of this model is expressed as:

$$Y(x) = \sum_{j=1}^{k} \beta_j f_j(x) + Z(x) \tag{24}$$

Where $f_j, j = 1, ..., k$ is assumed as a known vector of function, β_j is an unknown constant needed to estimated, and $Z(\cdot)$ is a stochastic process, commonly assumed to be Gaussian, with mean zero and covariance

$$\begin{aligned} Cov(Z(w), Z(u)) &= \sigma^2 R(w, u) \\ &= \sigma^2 \exp\left\{ -\theta \sum_{i=1}^{d} (w_i - u_i)^2 \right\} \end{aligned}$$

where σ^2 is the process variance. In practice, the linear model component in (20) is often reduced to only an intercept b since the inclusion of a more complex linear model does not necessarily yield a better prediction.

A.4 ε-SVR

Given the data set $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_l, y_l)\}$(where l denotes the number of samples) and the kernel matrix $K_{ij} = K(\mathbf{x_i}, \mathbf{x_j})$, and if the loss function in SVR is ε-insensitive loss function

$$L_\varepsilon\left(f(\mathbf{x}) - y\right) = \begin{cases} 0, & |f(\mathbf{x}) - y| < \varepsilon \\ |f(\mathbf{x}) - y| - \varepsilon, & other \end{cases}, \tag{25}$$

then the ε-SVR is written as:

$$\min\ \Phi(\mathbf{w}, \xi) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\left(\xi_i^- + \xi_i^+\right) \tag{26}$$

$$s.t. \begin{cases} f\left(\mathbf{x}_i\right) - y_i \leq \varepsilon + \xi_i^+ \\ y_i - f\left(\mathbf{x}_i\right) \leq \varepsilon + \xi_i^- \\ \xi_i^-, \xi_i^+ \geq 0, \end{cases}, \ i = 1, \cdots l.$$

The Lagrange dual model of the above model is expressed as:

$$\begin{aligned} \min_{\alpha^{(*)}} \frac{1}{2}\sum_{i,j=1}^{l}\left(\alpha_i - \alpha_i^*\right)\left(\alpha_j - \alpha_j^*\right) K\left(\mathbf{x}_i, \mathbf{x}_j\right) \\ - \sum_{i=1}^{l}\left(\alpha_i - \alpha_i^*\right) y_i + \varepsilon \sum_{i=1}^{l}\left(\alpha_i + \alpha_i^*\right) \end{aligned} \tag{27}$$

$$s.t. \begin{cases} 0 \leq \alpha_i, \alpha_i^* \leq C\ , i = 1, \cdots, l, \\ \sum_{i=1}^{l}\left(\alpha_i - \alpha_i^*\right) = 0. \end{cases},$$

where $K(\cdot, \cdot)$ is kernel function. After being worked out the parameter $\alpha^{(*)}$, the regression function $f(\mathbf{x})$ can be gotten.

Appendix B: Box plots

In a box plot, the box is composed of lower quartile (25%), median (50%), and upper quartile (75%) values. Besides the

box, there are two lines extended from each end of the box, whose upper limit and lower limit are defined as follows:

$$\text{low\_limit} = \max\{Q_1 - 1.5\text{IQR}, X_{\text{minimum}}\} \quad (28)$$

$$\text{up\_limit} = \min\{Q_3 + 1.5\text{IQR}, X_{\text{maximum}}\} \quad (29)$$

where Q_1 is the value of the line at lower quartile, Q_3 is the value of the line at upper quartile, IQR $= Q_3 - Q_1$, $X_{minimum}$ and $X_{maximum}$ are the minimum and maximum value of the data. Outliers are data with values beyond the ends of the lines by placing a "+" sign for each point.

References

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

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

Clarke SM, Griebsch JH, Simpson TW (2005) Analysis of support vector regression for approximation of complex engineering analyses. Trans ASME J Mech Des 127(6):1077–1087

Cresssie N (1988) Spatial prediction and ordinary kriging. Math Geol 20(4):405–421

De Boor C, Ron A (1990) On multivariate polynomial interpolation. Constr Approx 6:287–302

Fang H, Horstemeyer MF (2006) Global response approximation with radial basis functions. Eng Optim 38(4):407–424

Forrester AIJ, Keane AJ (2009) Recent advances in surrogate-based optimization. Prog Aerosp Sci 45(1–3):50–79

Friedman JH (1991) Multivariate adaptive regressive splines. Ann Stat 19(1):1–67

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

Hardy R (1971) Multiquadratic equations of topography and other irregular surfaces. J Geophys Res 76:1905–1915

Jin R, Chen W, Simpson TW (2001) Comparative studies of metamodeling techniques under multiple modeling criteria. Struct Multidisc Optim 23(1):1–13

Kleijnen JPC, Sanchez SM, Lucas TW, Cioppa TM (2005) A users guide to the brave new world of designing simulation experiments. INFORMS J Comput 17(3):263–289

Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Fourteenth international joint conference on artificial intelligence, pp 1137–1143

Langley P, Simon HA (1995) Applications of machine learning and rule induction. Commun ACM 38(11):55–64

McDonald D, Grantham W, Tabor W, Murphy M (2000) Response surface model development for global/local optimization using radial basis functions. In: The 8th AIAA symposium on multidisciplinary analysis and optimization, Long Beach, CA

Meckesheimer M, Barton R, Simpson T, Limayemn F, Yannou B (2001) Metamodeling of combined discrete/continuous responses. AIAA J 39(10):1950–1959

Meckesheimer M, Barton R, Simpson T, Booker A (2002) Computationally inexpensive metamodel assessment strategies. AIAA J 40(10):2053–2060

Papadrakakis M, Lagaros M, Tsompanakis Y (1998) Structural optimization using evolution strategies and neural networks. Comput Methods Appl Mech Eng 156(1–4):309–333

Perrone M, Cooper L (1993) When networks disagree: ensemble methods for hybrid neural networks. In: Mammone RJ (ed) Artificial neural networks for speech and vision. Chapman and Hall, London, pp 126–142

Picard R, Cook R (1984) Cross-validation of regression models. J Am Stat Assoc 79(387):575–583

Powell M (1987) Radial basis functions for multivariable interpolation: a review. In: Mason JC, Cox MG (eds) Proceedings of the IMA conference on algorithms for the approximation of functions and data. Oxford University Press, London, pp 143–167

Queipo NV, Haftka RT, Shyy W, Goel T, Vaidyanathan R, Tucker PK (2005) Surrogate-based analysis and optimization. Prog Aerosp Sci 41:1–28

Sacks J, Schiller SB, Welch WJ (1989a) Designs for computer experiments. Technometrics 31(1):41–47

Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989b) Design and analysis of computer experiments. Stat Sci 4(4):409–435

Simpson TW, Toropov V, Balabanov V, Viana FAC (2008) Design and analysis of computer experiments in multidisciplinary design optimization: a review of how far we have come or not. In: 12th AIAA/ISSMO multidisciplinary analysis and optimization conference, AIAA20085802, Victoria, BC, Canada

Viana FAC, Haftka RT, Steffen V (2009) Multiple surrogate: how cross-validation errors can help us to obtain the best predictor. Struct Multidisc Optim 39:439–457

Viana FAC, Gogu C, Haftka RT (2010) Making the most out of surrogate models: tricks of the trade. In: ASME 2010 international design engineering technical conferences and computers and information in engineering conference, DETC2010-8813, Montreal, Canada

Wang GG, Shan S (2007) Review of metamodeling techniques in support of engineering design optimization. Trans ASME J Mech Des 129(4):370–381

Yang Y (2003) Regression with multiple candidate models: selecting or mixing? Stat Sin 13(5):783–809

Zerpa L, Queipo N, Pintos S, Salager J (2005) An optimization methodology of alkaline-surfactant-polymer flooding processes using field scale numerical simulation and multiple surrogates. J Pet Sci Eng 47:197–208