An adaptive sampling method for global sensitivity analysis based on least-squares support vector regression

Maria Steiner[a,*]**, Jean-Marc Bourinet**[b]**, Tom Lahmer**[c]**,**

[a]*Chair of Computing in Civil Engineering, Bauhaus-Universität Weimar, Germany*
[b]*Institut Pascal, SIGMA Clermont, France*
[c]*Institute of Structural Mechanics, Bauhaus-Universität Weimar, Germany*

Abstract. In the field of engineering, surrogate models are commonly used for approximating the behavior of a physical phenomenon in order to reduce the computational costs. Generally, a surrogate model is created based on a set of training data, where a typical method for the statistical design is the Latin hypercube sampling (LHS). Even though a space-filling distribution of the training data is reached, the sampling process takes no information on the underlying behavior of the physical phenomenon into account and new data cannot be sampled in the same distribution if the approximation quality is not sufficient. Therefore, in this study we present a novel adaptive sampling method based on a specific surrogate model, the least-squares support vector regression. The adaptive sampling method generates training data based on the uncertainty in local prognosis capabilities of the surrogate model - areas of higher uncertainty require more sample data. The approach offers a cost efficient calculation due to the properties of the least-squares support vector regression. The opportunities of the adaptive sampling method are proven in comparison with the LHS on different analytical examples. Furthermore, the adaptive sampling method is applied to the calculation of global sensitivity values according to Sobol, where it shows faster convergence than the LHS method. With the applications in this paper it is shown that the presented adaptive sampling method improves the estimation of global sensitivity values, hence reducing the overall computational costs visibly.

---

* Corresponding author.
  E-mail addresses: maria.steiner@uni-weimar.de (M. Steiner)
                    jean-marc.bourinet@sigma-clermont.fr (J.-M. Bourinet)
                    tom.lahmer@uni-weimar.de (T. Lahmer)

1 Introduction

In numerous fields in civil engineering, numerical models and physical experiments representing the reality are applied for observing physical phenomenona. For improving modeling properties, fields like model calibration and uncertainty analysis are topics of various investigations [1–4]. Uncertainty analysis involves the determination of uncertainties in the model responses derived from uncertainties in the model parameters and of the relationships between the model parameters and the model responses, which is done by sensitivity analyses. There are a variety of approaches for uncertainty and sensitivity analysis in use, such as differential analysis [5–7], variance decomposition procedures [8–11], and sampling-based approaches [12–15]. In this study, the focus is on sampling-based approaches, which convince through easy implementation and the waiver of intermediate models. However, sampling-based approaches involve high computational cost because a high amount of data must be calculated with numerical models.

A commonly applied technique to reduce computational cost is to use data-based models, socalled surrogate models. Through that, not the full amount of the data needs to be simulated with the original model because the applied surrogate approximates parts of the data and the computation time can be effectively reduced. Surrogate models are frequently applied for sensitivity analysis [16–18], reliability analysis [19–21], and nonlinear optimization [22, 23]. This paper focuses in particular on the application of surrogate models to global sensitivity analysis. Two of the most popular surrogate models used for global sensitivity analysis are, for example, polynomial chaos expansions [24–27] and Kriging approximations [16, 28–30].

For sufficient results in the analyses, the approximation with the surrogate models must become sufficiently accurate using a limited number of data for training. An influence on the accuracy of the approximation has on the one hand the choice of the surrogate model and the corresponding model parameters and on the other hand the structure of the training data. This contribution deals with the second issue and especially with the challenge of adding new points to an existing sampling set. Therefore, we present a novel adaptive sampling method with the aim to accelerate and improve surrogate-based sensitivity analysis.

The paper is organized as follows. After a brief review of the main surrogate modeling techniques, we address issues of adaptive sampling strategies; then, we present the least-squares support vector regression, which is the applied surrogate model, and introduce the novel adaptive sampling method. Finally, we analyze the functionality and applicability of this method and observe the impact on global sensitivity analysis.

1.1 Surrogate modeling

The concept of constructing a surrogate model requires a set of training data points $\mathbf{x}_1, ..., \mathbf{x}_n$ within a domain $\mathscr{D} \subset \mathbb{R}^k$ with known responses $\mathbf{y} = [y_1, ..., y_n]^{\mathrm{T}} = [f(\mathbf{x}_1), ..., f(\mathbf{x}_n)]^{\mathrm{T}}$ of the observed model, a black box function f, which represents either a physical or a computer experiment. The behavior of the original function f, whose evaluation is usually time-intensive, can be approximated with a surrogate $\hat{f}$ based on the training data set. There are several techniques for constructing the approximation function and some intensively discussed are:

- Polynomial regression [23, 31]

- Moving least squares [32]

- Radial basis function [33, 34]

- Kriging regression [22, 35]

- Support vector regression (SVR) [36, 37]

- Artificial neural network [38].

All of these methods possess various properties and require different computational costs. Therefore, it is often a challenging question to judge on the optimal model choice. In the approach presented in this paper, we focus on the least-squares support vector regression (LS-SVR) because it provides the basis for the investigated method with a favorable calculation of the leave-one-out cross-validation error.

Besides the model selection, the construction of the training data set is a frequently discussed question. On one hand, there are different sampling possibilities to distribute the training points, such as full factorial and stratified random sampling techniques [22]. A stratified random sampling method commonly used is the Latin hypercube sampling (LHS), which we also apply for our analyses. On the other hand, the number n of training points has to be defined optimally. There is a trade-off between increasing the approximation quality, which mostly improves with more sample points, and keeping the computation cost low, which is related to choosing n small. Because the number of required training points is not known, it makes sense to apply adaptive sampling strategies, which is the topic of the next section.

1.2 Review of adaptive sampling methods

The LHS method provides a favorable, space-filling distribution of the training points. Nevertheless, if a sufficient approximation quality is not reached, new sample points have to be added to the existing training data set. This expansion cannot be done in the strategy of the LHS method. In the application fields of reliability analysis and nonlinear optimization, various adaptive sampling methods exist that sample new points regarding their objectives (e.g. [20, 39–41]). However, we are interested here in the application to global sensitivity analysis, which requires a good approximation quality over the whole range of the parameters. Therefore, areas with low approximation quality have to be identified and improved by adding new training points. Some existing global adaptive sampling strategies are listed here:

- Kriging approaches
  For example, the entropy approach [42]

$$\max_{\mathbf{x}_{\text{new}}} |\mathbf{R}_{n+1}| \left| \mathbf{1}_{n+1}^{\text{T}} \mathbf{R}_{n+1}^{-1} \mathbf{1}_{n+1} \right| \tag{1}$$

- Cross-validation approach [43]

$$\max_{\mathbf{x}_{\mathrm{new}}} \frac{1}{n} \sum_{i=1}^{n} \left(\hat{f}^{-i}(\mathbf{x}_{\mathrm{new}}) - \hat{f}(\mathbf{x}_{\mathrm{new}}) \right)^2 \tag{2}$$

- Maximin distance approach [44]

$$\max_{\mathbf{x}_{\mathrm{new}}} \left(\min_{i} \left(d(\mathbf{x}_{\mathrm{new}}, \mathbf{x}_i) \right) \right). \tag{3}$$

The Expressions 1, 2, and 3 describe the corresponding selection criteria for a new training point $\mathbf{x}_{\mathrm{new}}$. The Kriging approaches use the benefits of the ordinary Kriging method, which provides the estimation of the prediction error. In the mentioned example, the entropy approach, the aim is to maximize the amount of information that can be obtained from new sample points. There, $\mathbf{R}_{n+1}$ describes the correlation matrix between the initial training data set and a new point, and $\mathbf{1}_{n+1}$ is a $(n+1)$-dimensional vector of 1's. Even when several sequential approaches based on the Kriging method exist, they are not applicable to other surrogate models.

In contrast, the cross-validation approach can be applied to all kinds of surrogate models. It calculates a mean squared error based on the deviation between the surrogate function $\hat{f}$ for n training points and the sub-surrogates $f^{-i}(\mathbf{x})$ constructed with the training data set except the i'th point for $i = 1, ..., n$. Therefore, it requires more computational effort because n additional approximation functions have to be constructed.

In the third mentioned method, the maximin distance approach, the information obtained from the existing surrogate are not taken into account; instead, the aim is to maximize the Euclidean distance $d(\mathbf{x}_i, \mathbf{x}_j)$ between the sample point $\mathbf{x}_i$ and its nearest neighbor $\mathbf{x}_j$. Therefore, new points are sampled where no information from the original function exists and not where the approximation quality is perhaps lower.

In this contribution, we introduce a novel global adaptive sampling method, which offers a cost efficient calculation due to the properties of the underlying LS-SVR.

2 Proposed approach

In this section, we explain first the applied surrogate method and secondly, we introduce the novel global adaptive sampling method.

2.1 Least-squares support vector regression

One class of surrogate methods are the SVR methods, which have been originally introduced in the context of binary classification [45, 46] and then extended to regression methods [47]. A special form of the SVR methods is the LS-SVR [48, 49], which will be explained in this section. In order to give an overview about the functionality of the LS-SVR, we explain the idea first in the linear case and extend it then to a nonlinear approximation. Finally, the calculation of the model parameters is described.
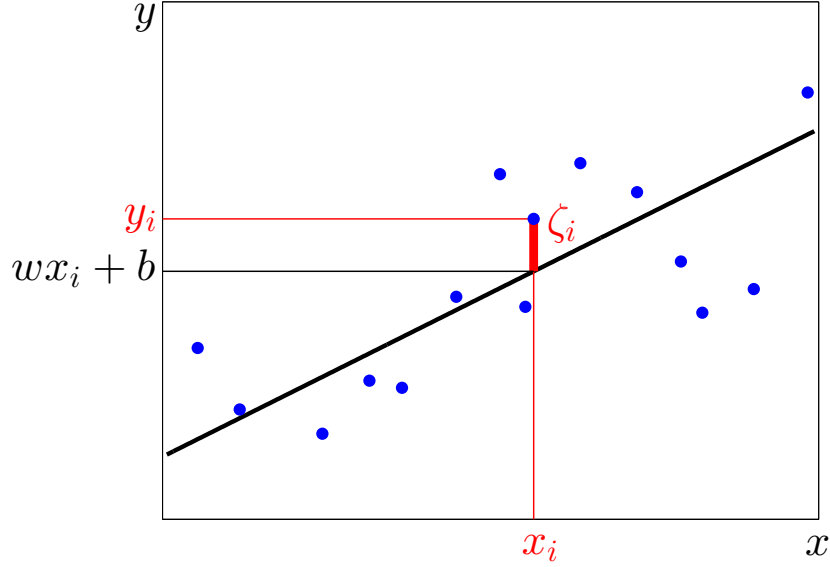
Figure 1: Linear least-squares SVR

Linear case:

The approximation function built in the linear case and expressed at point $\mathbf{x}$ is formulated as followed:

$$\hat{f}(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b, \tag{4}$$

where $\mathbf{w} = [w_1, ..., w_k]^{\mathrm{T}}$ and b are the unknown parameters of the method. To identify the optimal parameters, the optimization problem

$$\min_{\mathbf{w}, b, \zeta} \frac{\|\mathbf{w}\|^2}{2} + \frac{C}{2} \sum_{i=1}^{n} \zeta_i^2, \tag{5}$$

$$\text{s.t. } y_i = \mathbf{w}^{\mathrm{T}}\mathbf{x}_i + b + \zeta_i, \quad i = 1, ..., n$$

has to be solved. We minimize in the concept of the LS-SVR the sum of squares of the error between the response of the original function and the response of the approximation $\zeta_i = y_i - \hat{f}(\mathbf{x}_i)$ instead of a sum of the absolute error greater than ε as for the standard SVR based on the ε-insensitive loss function. The resulting advantage will be discussed later. The sum of squared errors is penalized by the parameter $C > 0$ to control the smoothness of the approximation. The linear approximation problem is displayed in Figure 1.

The optimization problem defined in Equation 5 can be reformulated into the following dual optimization problem by applying the Lagrangian function and the Karush-Kuhn-Tucker conditions for optimality:

$$\begin{bmatrix} \mathbf{K} + C^{-1}\mathbf{I}_{n\times n} & \mathbf{1}_n \\ \mathbf{1}_n^{\mathrm{T}} & 0 \end{bmatrix} \begin{pmatrix} \alpha \\ b \end{pmatrix} = \begin{pmatrix} \mathbf{y} \\ 0 \end{pmatrix} \tag{6}$$

with $(\mathbf{K})_{ij} = \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j$ and the n Lagrangian multipliers $\alpha = [\alpha_1, ..., \alpha_n]^{\mathrm{T}}$. Since this optimization problem consists of a linear set of equations, it can be solved easily to obtain the parameters of

the approximation function, expressed as

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i^{\mathrm{T}} \mathbf{x} + b. \tag{7}$$

Whereby, the construction of the approximation function is faster than for the classical SVR, which require solving a quadratic optimization problem. The only concern that is listed against the LS-SVR in comparison with the classical SVR, is that the prediction is written in terms of all training points and thereby cannot be formulated in terms of a sparse representation. This can lead to higher processing time if n gets high, which should be avoided already because of the computational costs of the original function f.

Nonlinear case:

The concept of the linear LS-SVR can be directly adapted to the nonlinear case by using kernel functions. The main idea is to map the input data into a nonlinear feature space denoted by $\mathscr{F}$, enabling a linear LS-SVR in $\mathscr{F}$. It is unnecessary to define the mapping $\phi : \mathscr{D} \to \mathscr{F}$ explicitly; choosing a kernel function k that corresponds to $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^{\top} \phi(\mathbf{x}_j)$ is sufficient [45, 50]. Therefore, the nonlinear approximation function is expressed as

$$\hat{f}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b. \tag{8}$$

For a linear mapping, Equation 8 becomes Equation 7. To identify the optimal model parameters α and b, the dual optimization problem from Equation 6 has to be solved, as for the linear case. Though, the entries of the matrix $\mathbf{K}$ are now defined by the kernel function with $(\mathbf{K})_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. Some possible choices for the kernel function are [37, 51]:

- Gaussian kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2})$ with $\sigma \in \mathbb{R}_{>0}$

- Polynomial kernel: $k(\mathbf{x}_i, \mathbf{x}_j) = (a + \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_j)^d$ with $d \in \mathbb{N}_{>0}$ and $a \in \mathbb{R}_{>0}$

- Sigmoid kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\kappa \mathbf{x}_i^{\mathrm{T}} \mathbf{x}_j - s)$ with κ and $s \in \mathbb{R}_{>0}$.

In this investigation we use the Gaussian kernel which is the most widely used kernel in the SVR literature.

Estimation of the model parameters:

To result in the most suitable approximation function, we need to identify the optimal choice of the kernel parameter σ and the regularization parameter C. If another kernel function is applied, different kernel parameters has to be identified accordingly. Therefore, we define the hyperparameters $\theta = [C, \sigma, ...]^{\mathrm{T}}$ including all utilized parameters.

As a criteria to decide about the parameter selection the K-fold cross-validation error [52] is frequently applied. Therefore, the training data set is split randomly into K sub data sets and sub-surrogates are built, for each of which one subset of the training points is left out. The

left-out subset can be used as an untrained test data set to examine the approximation quality in unobserved areas. The smaller K is choosen, the faster is the calculation of the cross-validation; however, the training data set gets quite reduced compared to the whole one and the variance of the cross-validation error increases. We apply in this study the special case $K = n$, known as leave-one-out error. The leave-one-out error is an almost unbiased estimate of the expected error [53] and provides therefore the best decision criterion. Because the calculation of the leave-one-out error is costly, commonly, a 10-fold cross-validation error is used [54]. However, by using the LS-SVR, the calculation cost of the leave-one-out error can be reduced, which is shown below.

For the leave-one-out error, the sub-surrogate $\hat{f}^{-i}$ is constructed each time with the training data set except the i'-th training point. The result of the sub-surrogate at the left-out point $\mathbf{x}_i$ is then compared with the result of the true function value y_i, so that the leave-one-out error Err_{LOO} is formulated as

$$Err_{\text{LOO}} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{f}^{-i}(\mathbf{x}_i, \boldsymbol{\theta}))^2. \tag{9}$$

By minimizing Err_{LOO}, we obtain the best choices for the hyperparameters $\boldsymbol{\theta}$. For the calculation of this error value, it is necessary to construct the corresponding sub-surrogates $\hat{f}^{-i}$ for $i = 1, ..., n$, which requires more computational time. However, the LS-SVR enables an analytical calculation of the deviation of the true responses y_i to the result of the sub-surrogates $\hat{f}^{-i}(\mathbf{x}_i)$ [55] through

$$y_i - \hat{f}^{-i}(\mathbf{x}_i) = \frac{\alpha_i}{(\tilde{\mathbf{K}}^{-1})_{ii}}, \tag{10}$$

where $(\tilde{\mathbf{K}}^{-1})_{ii}$ is the i-th diagonal element of the inverse of

$$\tilde{\mathbf{K}} = \begin{bmatrix} \mathbf{K} + C^{-1}\mathbf{I}_{n \times n} & \mathbf{1}_n \\ \mathbf{1}_n^{\text{T}} & 0 \end{bmatrix}. \tag{11}$$

With the analytical formulation of the leave-one-out error, the LS-SVR offers a very efficient parameter estimation. Therefore, and due to the linear problem settings, the LS-SVR is a practical method with low computational costs, especially useful for high dimensional problems. Furthermore, the analytical formulation of the leave-one-out error offers the basis for the adaptive sampling method explained in the next section.

2.2 Distance-based LOO error sampling method

The global adaptive sampling method we introduce in this investigation is called the "distance-based LOO error sampling method". It uses two different criteria to choose suitable new sampling points.

First, we want to identify areas which have a higher uncertainty in the approximation quality than others. An indicator for the uncertainty of the whole approximation function is the leave-one-out error defined in Equation 9. This error consists of the sum of the deviation at each training point, so that each summand describes the uncertainty at one training point. In case of the LS-SVR, the analytical formulation of each of these deviations is given in Equation 10.

The uncertainty of the approximation function at an unobserved point can be approximated by the uncertainty at the training points, implying that the behavior of the model uncertainty is continuous. More precisely, we use the deviations $y_i - \hat{f}^{-i}(\mathbf{x}_i)$ for $i = 1, ..., n$ as responses of the training points to construct the leave-one-out error function $\hat{f}_{\text{LOO}}(\mathbf{x})$ by applying the LS-SVR. New sampling points can be then obtained by maximizing the function $\hat{f}_{\text{LOO}}$ over $\mathbf{x}$.

Second, we want to avoid getting redundant information by constructing new training points. However, it is possible that $\hat{f}_{\text{LOO}}$ is maximal at or close to an existing training point. Therefore, we take additionally the maximin distance approach into account, which is also done in the cross-validation approach in [43]. The maximin distance approach prevents close and therefore redundant training points.

By taking both criteria into account, we obtain an optimization problem where we have to maximize both the leave-one-out error function $\hat{f}_{\text{LOO}}$ and the minimal distance between the sample points. This optimization problem can be formulated as follows:

$$\max_{\mathbf{x} \in \mathscr{D}} (\hat{f}_{\text{LOO}}(\mathbf{x}) \min_i d(\mathbf{x}, \mathbf{x}_i)). \tag{12}$$

As a distance function $d(\mathbf{x}, \mathbf{x}_i)$ the Euclidean metric is selected.

Selection of suitable points:

Still, there is the question how to solve the optimization problem from Expression 12. We use the following two ways to select new sampling points related to this optimization problem:

- Selection out of a predifined large sample set $\mathbf{X}^{\text{new}}$ with e.g. $N_{\text{new}} = 50\,000$

- Sampling related to the quasi-distribution function $\delta(\mathbf{x})$ with $\delta(\mathbf{x}) = \hat{f}_{\text{LOO}}(\mathbf{x}) \min_i d(\mathbf{x}, \mathbf{x}_i)$.

In the first case, new points are taken from a large sample set. We choose the points with the highest result for $\hat{f}_{\text{LOO}}(\mathbf{x}) \min_i d(\mathbf{x}, \mathbf{x}_i)$. This calculation needs a computational effort depending on the size N_{new} of the sample set where we choose the new points from. Therefore, it is recommended to use this way to select new points only for research analyses. For numerical and physical applications, where small computation time is desired, the second selection method is recommended. There, the objective of the optimization problem is taking as a quasi-distribution function $\delta(\mathbf{x})$. New points are randomly sampled along $\delta(\mathbf{x})$ with the slice sampling routine [56]. Accordingly, sample points with higher results for the optimization function have a higher possibility to be chosen. After identifying the new sample points with one of the two methods the corresponding responses have to be calculated with the original function. With the expanded training data set, a new surrogate can be constructed.

In the following, we explain in more details how the adaptive sampling algorithm works and give a pseudo-code for a Matlab implementation.

Implementation:

In the main program, first, the LS-SVR is constructed with the initial training data set $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_n]^{\text{T}}$ and their responses of the original function $\mathbf{y}$. During the approximation process

the parameters α and b are calculated by solving Equation 6, and θ by minimizing Equation 9. We also obtain the results $y_i - \hat{f}^{-i}(\mathbf{x}i)$ from Equation 10 for $i = 1,...,n$, which are used for the adaptive sampling method. As the second part stepwise new points are added to $\mathbf{X}$ and $\mathbf{y}$ by applying the distance-based LOO error sampling method. The number m of calls of the adaptive sampling method and the number n_{new} of new points in each step is predefined. However, it is possible to include a termination criterion in the corresponding loop such as Err_{LOO} greater than a constant number corresponding to a sufficient approximation quality. Therefore, the limit for sufficient approximation quality is strongly dependent on the observed problem; influencing factors could be the dimension, parameter range, assumed noise in the input parameters, or the application field. The pseudo-code of the main program is listed below.

Main program
% Constructs the approximation function hf based on the LS-SVR with the use of the distance-based LOO error method

% Input: X – initial training data set (size: n × k)
% y – responses of the training set (= f(X))
% m – number of steps for the adaptive sampling strategy
% n_new – number of new points in each step

%% Calculation of the parameters for the LS-SVR:
[SVR_par, LOOerr_vec] = LS_SVR_parameter (X, y);
    % SVR_par – Parameter of the LS-SVR: α, b, θ $(= [C, \sigma, ...])$
    % calculated by solving Equation 6 and minimizing Equation 9
    % LOOerr_vec – results from Equation 10 for $i = 1,...,n$

%% Adaptive sampling:
for i = 1 : m
    Adaptive_sampling % chooses the new training points x_new
    y_new = f(x_new); % new responses from the original function
    %% Update of the parameters:
    X = [X; x_new]; y = [y; y_new];
    n = n + n_new;
    [SVR_par, LOOerr_vec] = LS_SVR_parameter (X, y);
end

%% Construction of the LS-SVR:
hf = @(x) LS_SVR_approximation (x, X, SVR_par);
    % hf – approximation function from Equation 8.

As explained earlier, there are two possibilities how to choose new sampling points in order to maximize Expression 12. First we explain the procedure where new points are picked out of a predefined sample set $\mathbf{X}_{\text{new}}$. It is necessary to compute the minimum distance of each point of $\mathbf{X}_{\text{new}}$ to the existing training data set $\mathbf{X}$ and the results of the function $\hat{f}_{\text{LOO}}$. Thereby, the results of $\hat{f}_{\text{LOO}} \min_i d(\mathbf{x}, \mathbf{x}_i)$ for all $\mathbf{x}$ in $\mathbf{X}_{\text{new}}$ can be calculated and compared. After a new point is chosen, just the minimum distances have to be adjusted with the new training point before

the next point can be chosen. Details are given in the following pseudo-code.

Adaptive_sampling (based on a large sample set)
% Samples new training points by maximizing Expression 12

% Input: X_new   – set of possible training points (size: N_new×k)
%   MD_new   – minimum distances of each entrance of X_new
%   to the existing training points
%   X   – previous training set
%   LOOerr_vec   – results from Equation 10
%   n_new   – number of new points

%% Determine the optimality criterion:
d = @(x, y) **sqrt**(**sum**((x-y).^2, 2);   % Euclidean distance
fLOO_par = LS_SVR_parameter (X, LOOerr_vec);
fLOO = @(x) **abs**(LS_SVR_approximation (x, X, fLOO_par));
      % leave-one-out error function
fLOO_new = fLOO(X_new);

%% Choose new points:
x_new = **zeros**(n_new, k);
for i = 1 : n_new
    [e, ie] = **max**(fLOO_new.∗MD_new); % maximizing Expression 12
    x_new(i, :) = X_new(ie,:);   % new sampling point
    %% Update of the parameters:
    X_new(ie, :) = [ ]; N_new = N_new - 1;
    MD_new(ie) = [ ]; fLOO_new(ie) = [ ];
    for i_N = 1 : N_new
        MD_new(i_N) = **min**(MD_new(i_N), d(X_new(i_N, :), x_new));
    end
end

For the second way to choose new sampling points, the slice sampling routine [56] is used. There, points are sampled along a pseudo-distribution function p. As an initial point a **x**-value is chosen with $p(\mathbf{x}) > 0$. Outgoing from the result from the previous step a random value u_i between 0 and $p(\mathbf{x}_{i-1})$ is chosen. Then, $\mathbf{x}_i$ is sampled randomly until a value is found with $p(\mathbf{x}_i) > u_i$. With this procedure sampling points with higher values of p have a higher probability to be selected. The one-dimensional case of one step of the slice sampling routine is shown in Figure 2. In our application $p(\mathbf{x})$ is chosen as $\hat{f}_{\text{LOO}} \min_i d(\mathbf{x}, \mathbf{x}_i)$. The corresponding adaptive sampling routine is presented below. In this program we assume the sample points in the interval $[0, 1]$, which has to be adjusted according to the problem of interest.
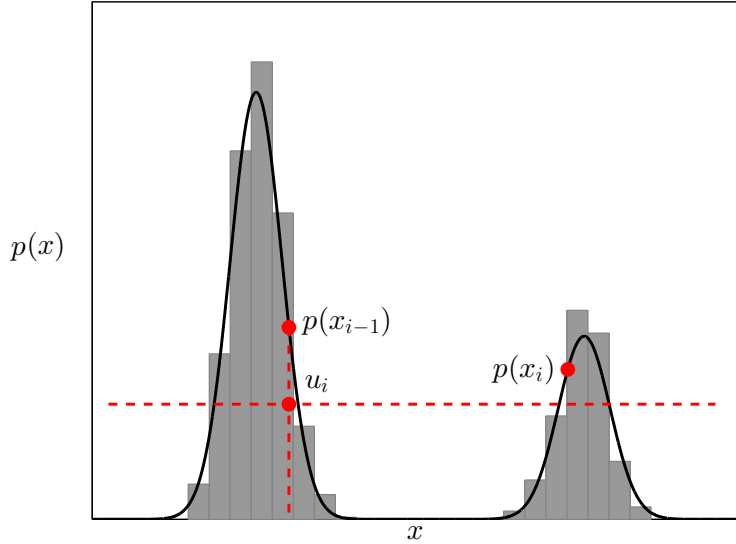
Figure 2: Slice sampling method

Adaptive_sampling (based on the slice sampling method)

% Input: X   – previous training set
%   LOOerr_vec   – results from Equation 10
%   n_new   – number of new points

%% Create the pseudo-probability function:
d = @(x,y)**sqrt**(**sum**((x-y).^2, 2)   % Euclidean distance
fLOO_par = LS_SVR_parameter (X, LOOerr_vec);
fLOO = @(x) **abs**(LS_SVR_approximation (x, X, fLOO_par));
p = @(x) fLOO(x)∗**min** d(x, X);   % pseudo-probability function

x_new = **zeros**(n_new, k);
xi = rand(1,k); pi = p(xi);   % start point for slice sampling

%% Slice sampling method:
for i = 1 : n_new
    ui = rand∗pi;
    xi = rand(1, k); pi = p(xi);
    while pi < ui
        xi = rand(1, k); pi = p(xi);
    end
    x_new(i, :) = xi
end

In the next section, we show with different examples the functionality and advantages of the distance-based LOO error sampling method.

3 Numerical analysis

In the following numerical analysis, we observe the efficiency of the introduced adaptive sampling method by comparing it with the LHS method. Although, the LHS method is not an adaptive sampling method, still it provides an equally distributed sample set better than the maximum distance approach; therefore, it gives a good comparison to the adaptive sampling method, where we improve the approximation quality in specific areas.

Two quality criteria are used to assess the approximation quality of the respective surrogate model. Both are based on the coefficient of determination (CoD) [57], which is formulated as followed

$$
\text{CoD} = 1 - \frac{\sum\limits_{i=1}^{n} \left(y_i - \hat{f}(\mathbf{x}_i)\right)^2}{\sum\limits_{i=1}^{n} \left(y_i - \left(\frac{1}{n} \sum\limits_{i=1}^{n} y_i\right)\right)^2}.
\tag{13}
$$

The CoD describes how much of the behavior of the original function can be described by the approximation and indicates a good approximation with a value close to 1. Although, the CoD is a meaningful error criterion, it uses the training points also as test points and could result therefore in misleading outcomes. Instead, we use on the one hand the cross-validation approach and on the other hand an additional unused data set. By using $\hat{f}^{-i}$ instead of $\hat{f}$, we result in the CoD based on cross-validation (CoD_{CV}). In the second case -the validated CoD (CoD_{val})- the test points x_i and their responses y_i with $i = 1,...,N$ come from a large data set (for example $N = 100\,000$), which was not used before. In the investigation, the most meaningful is the CoD_{val}, but it can be used only for selected examples where the amount of available data is large enough.

In the following subsections, we observe different functions which represent cases that can appear in real-world problems and investigate the application of the adaptive sampling method to the sensitivity analysis. For a better overview, the discussed examples are listed hereinafter:

- One-dimensional model $f(x) = x \sin x$ (Subsection 3.1)

- Three-dimensional model $f(\mathbf{x}) = x_1 \sin x_1 + x_2 \sin x_2 + x_3 \sin x_3$ (Subsection 3.2)

- Model with Singularity $H(\omega) = 1/(-m\omega^2 + c\omega i + k)$ (Subsection 3.3)

- Model with decaying influence of the variables $f(\mathbf{x}) = \sum\limits_{i=1}^{k} \frac{x_i}{\sqrt{i}} \sin \sqrt{i} x_i$ (Subsection 3.4)

- Noisy function (Subsection 3.5)

With the first example we show how the introduced adaptive sampling method works. Afterwards, the aim is to show the efficiency of the distance-based LOO error sampling method by comparing the results of different function types represented by the listed examples. There, we start for the application of the adaptive sampling method with a LHS data set of size $n = 5$ (if

$k = 1$) or $n = 25$ (if $k > 1$) and add each step 1 or 5 points accordingly in the sense of the adaptive sampling method. Based on this constructed training data set a LS-SVR is built and then tested with the CoD_{CV} and the CoD_{val}. We compare this results with the calculations based on full LHS data sets. To get an overview on the expected quality and the variance of the approximation, 100 independent simulations are performed and results are displayed in boxplots. The mean 50% of the data are represented by a box with the middle line as the median. With the whiskers, 95% of the results are depicted and the remaining are outliers.

3.1 One-dimensional model $f(x) = x \sin x$

As a first example showing the functionality of the distance-based LOO error method, we observe the approximation of the one-dimensional function $f(x) = x \sin x$ within the interval $[0, 15]$. In Figure 3, on the left hand side, the approximations with six, seven, and eight training points are plotted, while on the right hand side, $\hat{f}_{\text{LOO}}$, $10 \min_i d(\mathbf{x}, \mathbf{x}_i)$, and $\hat{f}_{\text{LOO}} \min_i d(\mathbf{x}, \mathbf{x}_i)$ are depicted. The new elected points regarding the optimization problem from Expression 12 are marked with a cross in all plots.

It is obvious that the new sampling points are selected in areas where the approximation quality has to be improved. Both optimization criteria affect the choice of new points. Often the $\hat{f}_{\text{LOO}}$ takes the maximum value close to already-existing training points. Therefore, including the maximin distance approach is important for the sampling strategy and depending on the existing approximation has more or less influence.

The function $f(x) = x \sin x$ is a function with similar behavior over the whole domain. It varies over the whole range; even though, the variation of the function increases with increasing x. Therefore, it is expected that an equally distributed sample set is of advantage and that the application of the adaptive sampling method cannot effectively improve the approximation quality. From the results of the CoD_{CV} and the CoD_{val} shown in Figure 4, it is visible that there is no great differences between the results of the adaptive sampling method and the LHS method. Nevertheless, the results indicate already some improvements by the use of the adaptive sampling method. By applying the distance-based LOO error sampling method, better results are obtained for $n = 7$ and $n = 9$. Also for higher n, the CoD_{val} of the adaptive sampling method is slightly larger than for the LHS method and the variance is smaller.

3.2 Three-dimensional model $f(\mathbf{x}) = x_1 \sin x_1 + x_2 \sin x_2 + x_3 \sin x_3$

As a second example we observe the three-dimensional function $x_1 \sin x_1 + x_2 \sin x_2 + x_3 \sin x_3$ within the interval $[0, 15]^3$. The variation of the function is similar to the previous one. Though, in this case more input parameter are important and have the same influence on the response of the function.

In Figure 5, the CoD_{CV}- and CoD_{val}-values are depicted. The results of the CoD_{CV} indicate clearly that the LHS method gives the better results. However, the CoD_{val} show that the adaptive sampling method improves the results from $n = 175$. An explanation is that, first sample points are chosen over the whole domain to adopt global behavior; later the local approximation quality
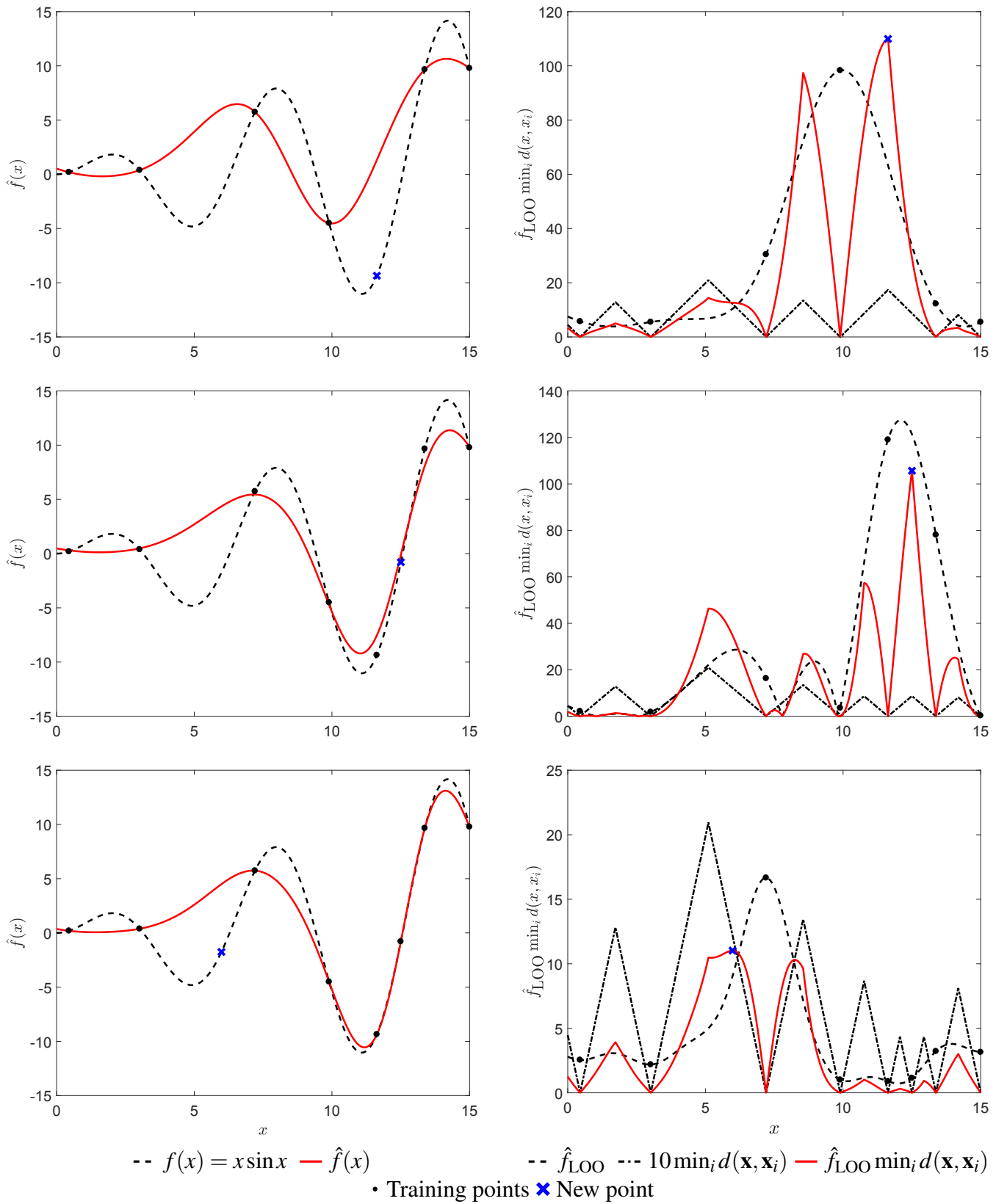
13

Figure 3: Choosing a new point with the distance-based LOO error method for $f(x) = x \sin x$, shown with the approximation function $\hat{f}(x)$ of $f(x)$ with six, seven, and eight points and the related $\hat{f}_{\text{LOO}} \min_i d(\mathbf{x}, \mathbf{x}_i)$
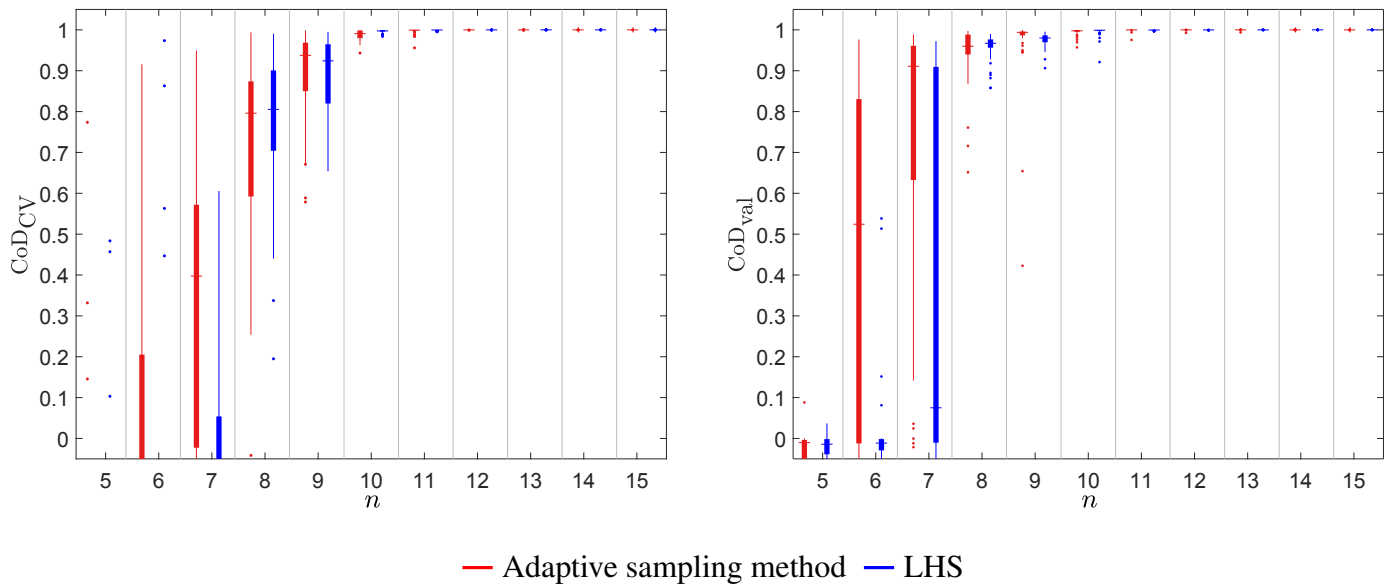
Figure 4: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of $x \sin x$
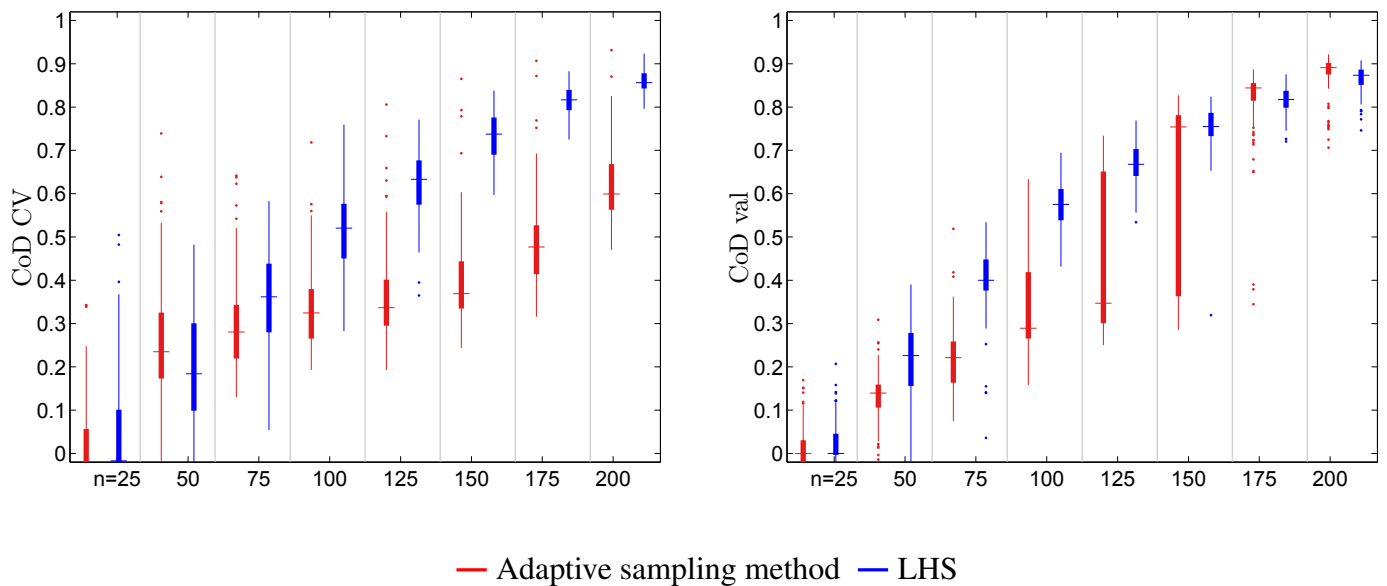

Figure 5: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of $x_1 \sin x_1 + x_2 \sin x_2 + x_3 \sin x_3$

can be improved. Therefore, the adaptive sampling method shows its improvement just for higher values of n and it would make sense to start later with this method.

It should be mentioned that the results of the CoD_{CV} do not show an improvement of the adaptive sampling method for any number of n. However, the CoD_{val} uses an additional data set and is therefore more trustable.

3.3 Model with Singularity $H(\omega) = 1/(-m\omega^2 + c\omega i + k)$

We are interested here in functions with weak discontinuities in some regions. This functions have some areas with low and high variations. As an example we study a typical function from engineering: The absolute value of the transfer function of the equation of motion. The problem is described as follows:

- Equation of motion: $m\ddot{u}(t) + c\dot{u}(t) + ku(t) = 0$
  Parameters: mass m, damping c, stiffness k,
  displacement $u(t)$, velocity $\dot{u}(t)$, acceleration $\ddot{u}(t)$ depending on the time t

- Transfer function: $H(\omega) = 1/(-m\omega^2 + c\omega i + k)$

- Observed function: $f(x) = |H(\omega)|$

- Parameters: $m = 1\,\text{kg}$, $c = 0.5\,\frac{\text{Ns}}{\text{m}}$, $k = 1\,\frac{\text{N}}{\text{m}}$.

The choice of the damping parameter c influences the damping effect and therefore the nonlinearity of the observed function. In Figure 6 we can see on the left hand side an approximation with the training points sampled using the adaptive sampling method and on the right hand side an approximation with LHS training points. It is evident that the first approximation is closer to the true function, because more sample points are constructed where the curvature of the graph of $f(x)$ changes more rapidly.

By comparing the results of the CoD_{val} for both the adaptive sampling method and the LHS method, it is also visible that the convergence to an acceptable solution is faster and more robust if the distance-based LOO error method is used, as illustrated in Figure 7. Again, the CoD_{CV} misinterprets the results for the smallest n and recommends the use of the LHS method; however, from $n = 12$ also the CoD_{CV} shows that the application of the adaptive sampling method provides better results.

3.4 Model with decaying influence of the variables

Another type of functions worth to investigate are functions where the input parameters have different influence on the output. Therefore, we analyze the following benchmark function

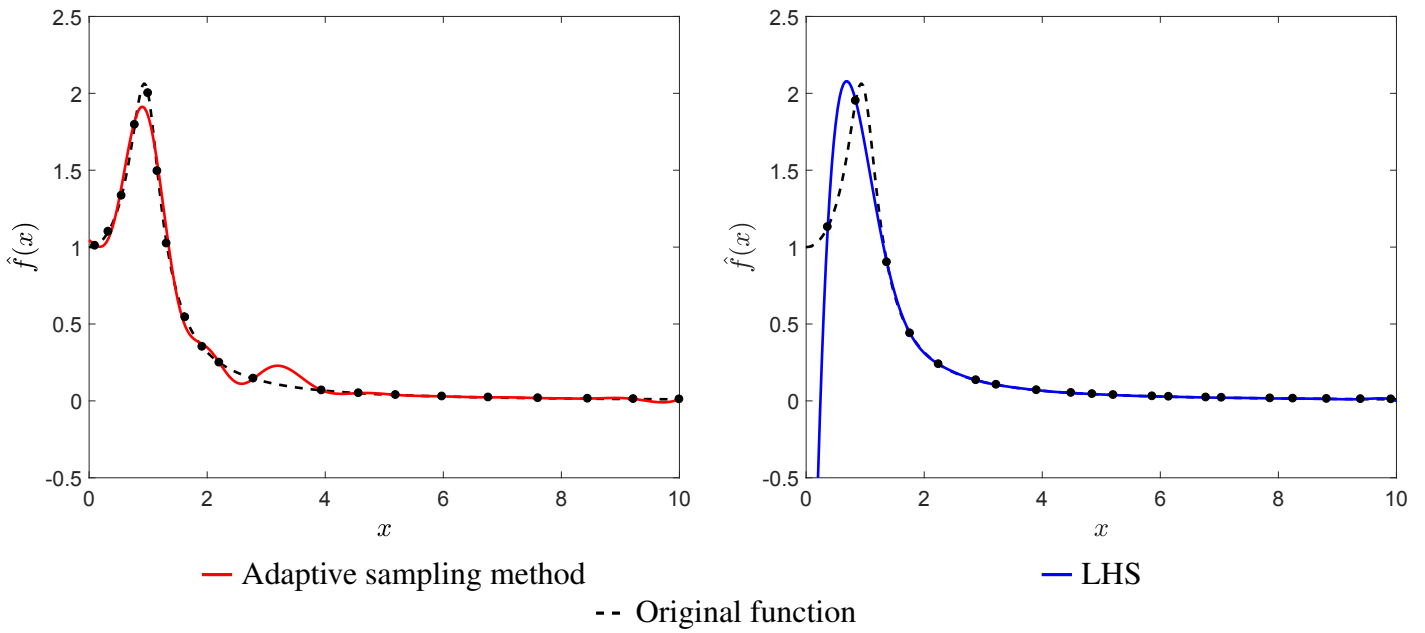$$f(\mathbf{x}) = \sum_{i=1}^{k} \frac{x_i}{\sqrt{i}} \sin \sqrt{i} x_i \tag{14}$$

16

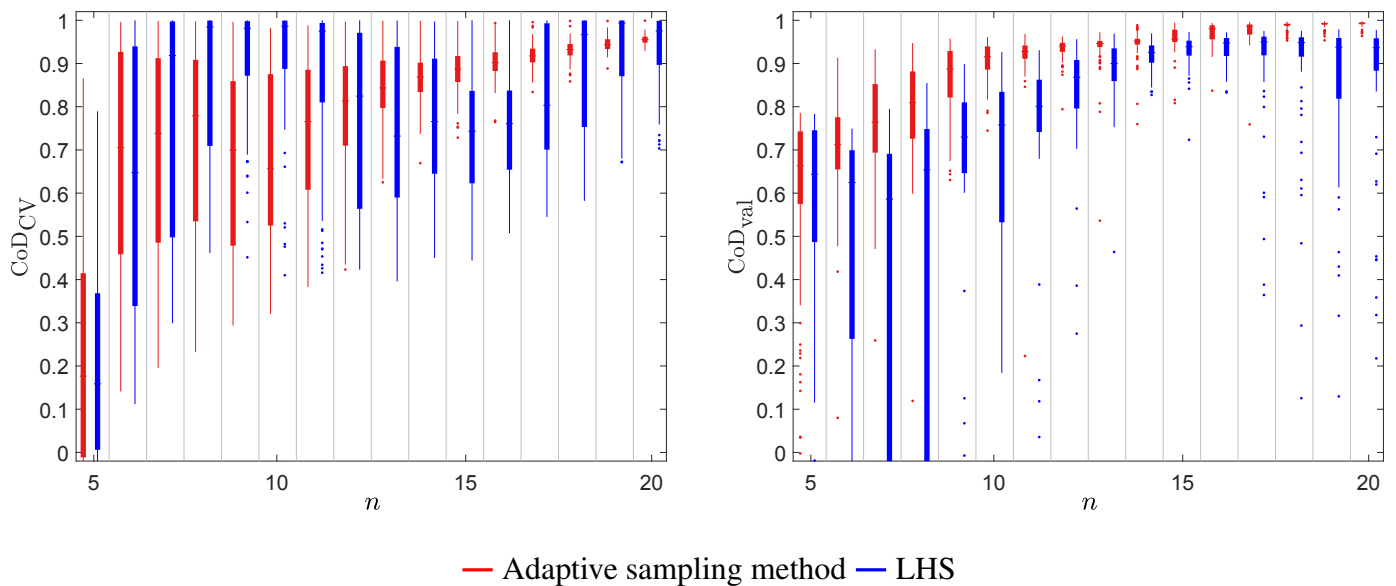Figure 6: Approximation with the adaptive sampling method and the LHS method with $n = 20$
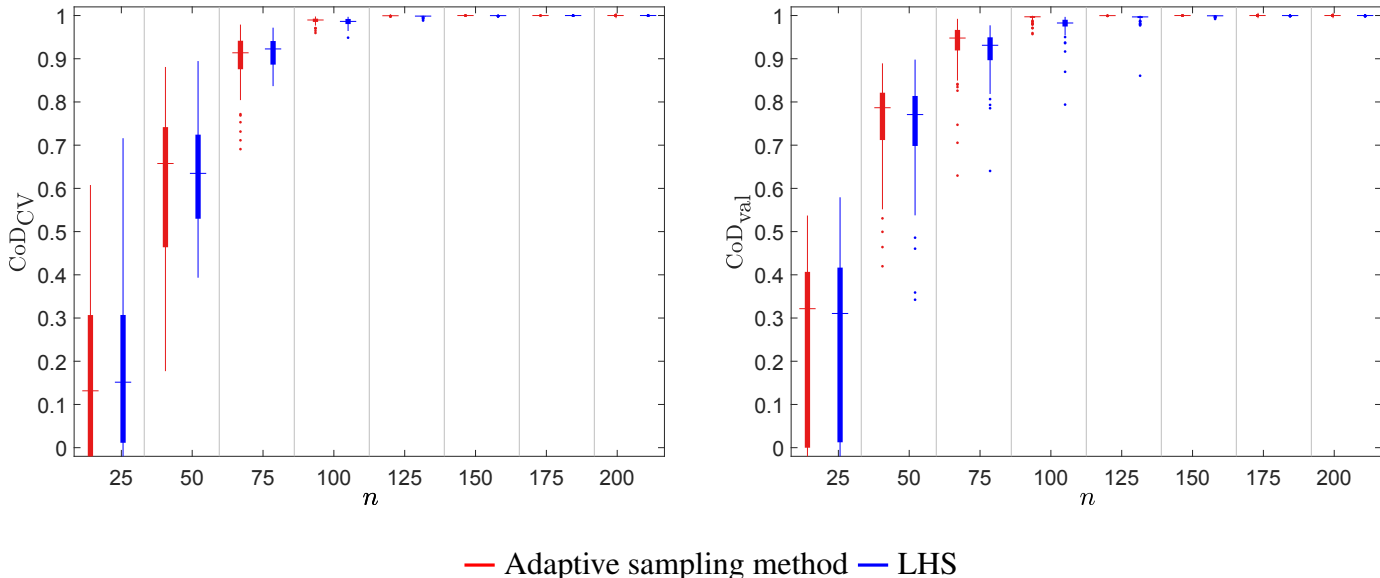

Figure 7: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of the absolute transfer function

within the interval $[0, 15]^k$. For increasing dimension, the input parameters with higher indices have less and less influence and we investigate if the adaptive sampling method can identify the different influences.

The result of the function f from Equation 14 for dimension $k = 2$ are displayed in Figure 8. From the values of the CoD_{CV} and the CoD_{val} it is visible that the approximation is already satisfying with $n = 100$ training point, but slightly better for the adaptive sampling method.

Figure 8: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of $x_1 \sin x_1 + \frac{x_2}{\sqrt{2}} \sin \sqrt{2} x_2$

In the three-dimensional case of the function f from Equation 14, we have to use a higher number of training points to get a sufficient approximation quality ($\text{CoD}_{\text{val}} \geq 0.9$). The results are shown in Figure 9. The values of the CoD_{CV} recommend to use only LHS data because the results of the distance-based LOO error sampling method are lower and have a higher variance. However, the CoD_{val} show that the approximation quality with the use of the adaptive sampling method is better for n larger than 300. Because at the beginning the adaptive sampling method not necessarily provides better approximation quality, it is recommended to start later sampling new points with respect to that strategy, depending on the dimension of the observed problem. For this example, it is recommended to start with $n = 200$ LHS training points and increase then the quality by adding new points regarding the distance-based LOO error method.

3.5 Noisy function

If the observed data are obtained from experiments, it is expected that the responses are corrupted by noise. In those cases it is desirable that the surrogate model approximates the behavior
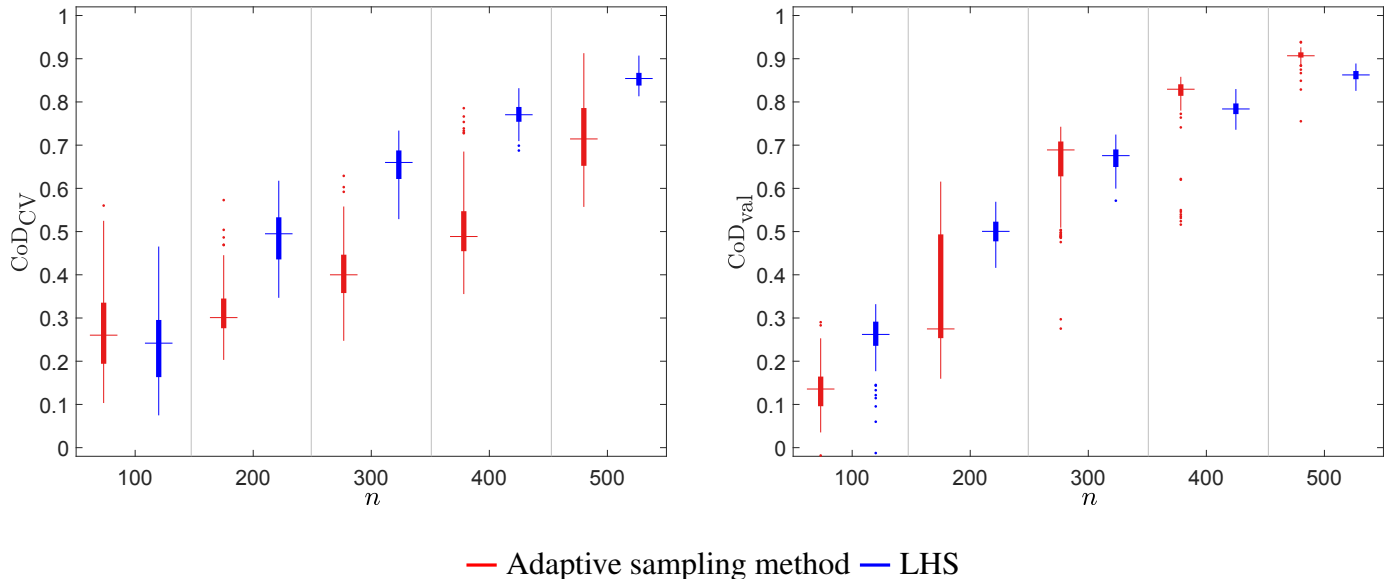
Figure 9: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of $x_1 \sin x_1 + \frac{x_2}{\sqrt{2}} \sin \sqrt{2}x_2 + \frac{x_3}{\sqrt{3}} \sin \sqrt{3}x_3$

of the underlying engineering effect and does not get lost in the noise. Especially for the adaptive sampling method, the noise should not prevent sampling new points in interesting areas.

To analyze noise corrupted functions, we revisit the model from Subsection 3.3 and add randomly to each sample point normally distributed noise with zero mean and a variance of 0.1 and 0.2. The corresponding approximations with and without the adaptive sampling method are displayed in Figure 10 and 11.

The higher the deviation of the training data from the original trend, the higher variation in both approximations is visible especially in the area that was nearly linear before. Still, the use of the adaptive sampling method offers a better approximation especially in the interesting area where the damping effect is visible. In the case where only the LHS method is applied, high deviations close to the edges of the domain can appear. Nevertheless, the displayed approximations are just possible surrogates, which change with the use of different training data. Therefore, we show in Figure 12 and 13 the results of 100 approximations with and without the use of the adaptive sampling method for the absolute transfer function with an additional noise distributed with $\mathcal{N}(0,0.1)$ and $\mathcal{N}(0,0.2)$.

With the displayed results of the CoD_{val} it is visible that with increasing noise the maximal possible approximation quality decreases, which is reasonable because the error in the training data set through the noise cannot be avoided. In the case with $\sigma = 0.1$ the variance of the CoD_{val} is higher for $n = 18, 19$ and 20 than for $n = 14$ to $n = 17$ if a LHS training data set is used. It shows that not always the use of more training points improves the approximation quality, rather the locations of new points are important. This effect is not visible for the higher noise level. From the results it can be concluded that the distance-based LOO error method provides better
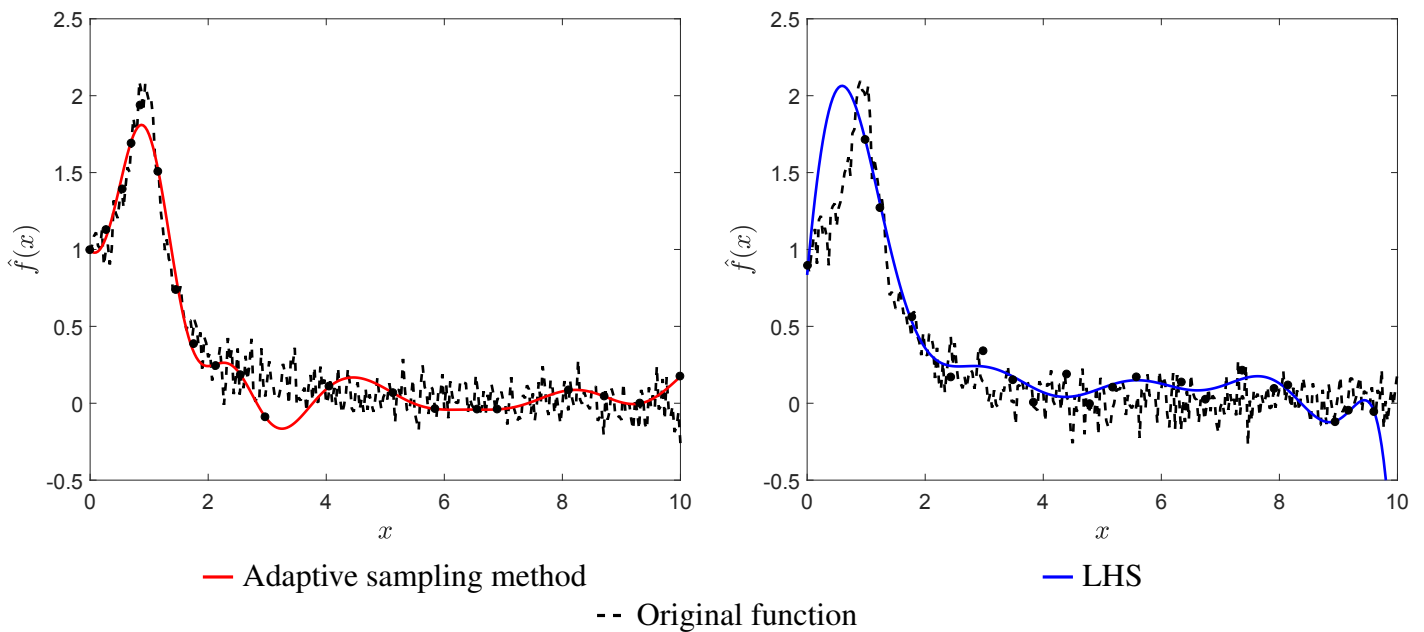
Figure 10: Approximation with the adaptive sampling method and the LHS method of the absolute transfer function with noise $\mathcal{N}(0, 0.1)$
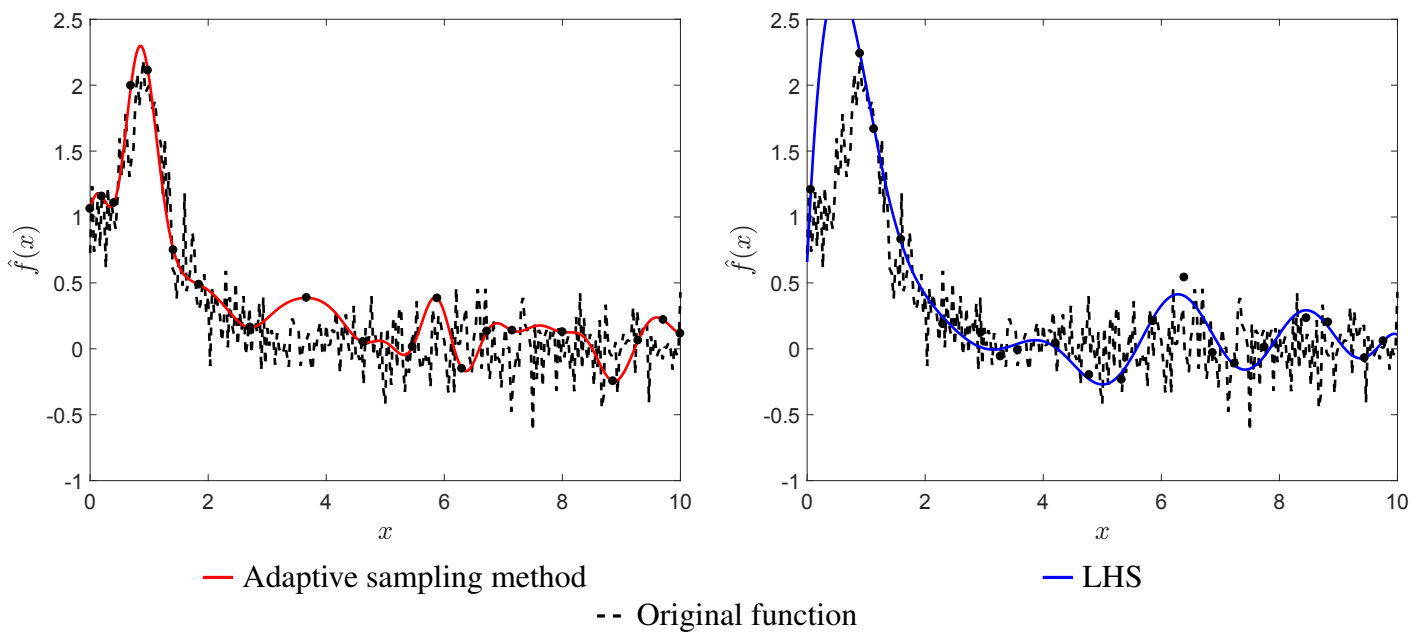

Figure 11: Approximation with the adaptive sampling method and the LHS method of the absolute transfer function with noise $\mathcal{N}(0, 0.2)$
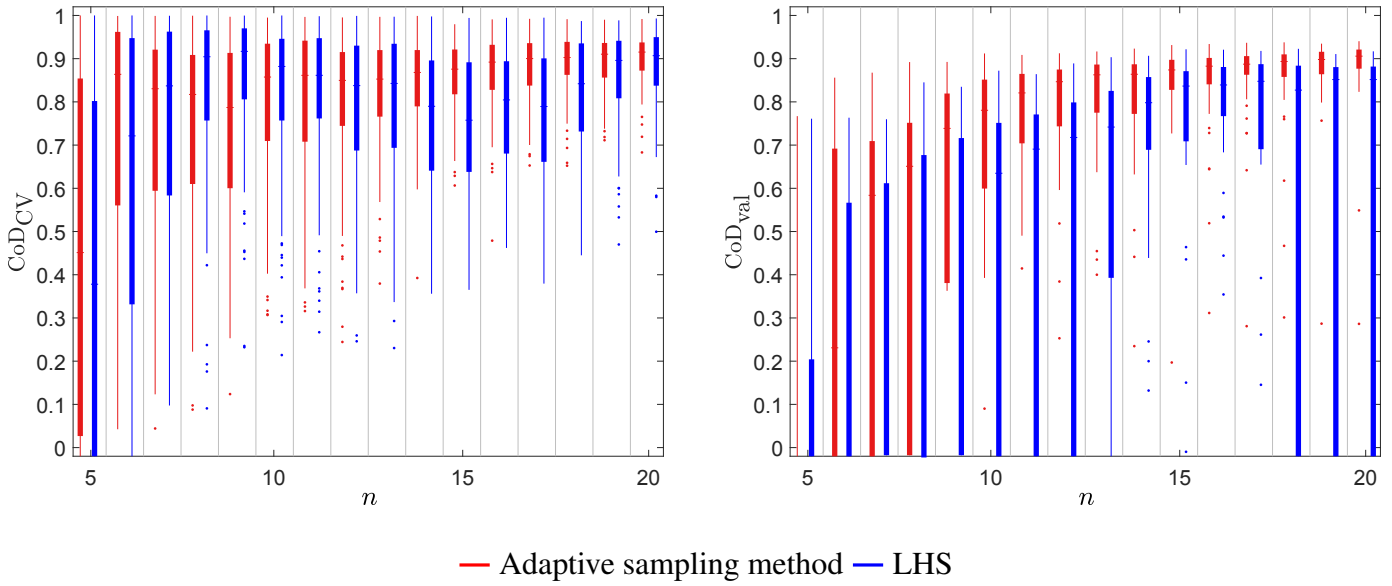
Figure 12: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of the absolute transfer function corrupted by noise distributed with $\mathcal{N}(0, 0.1)$
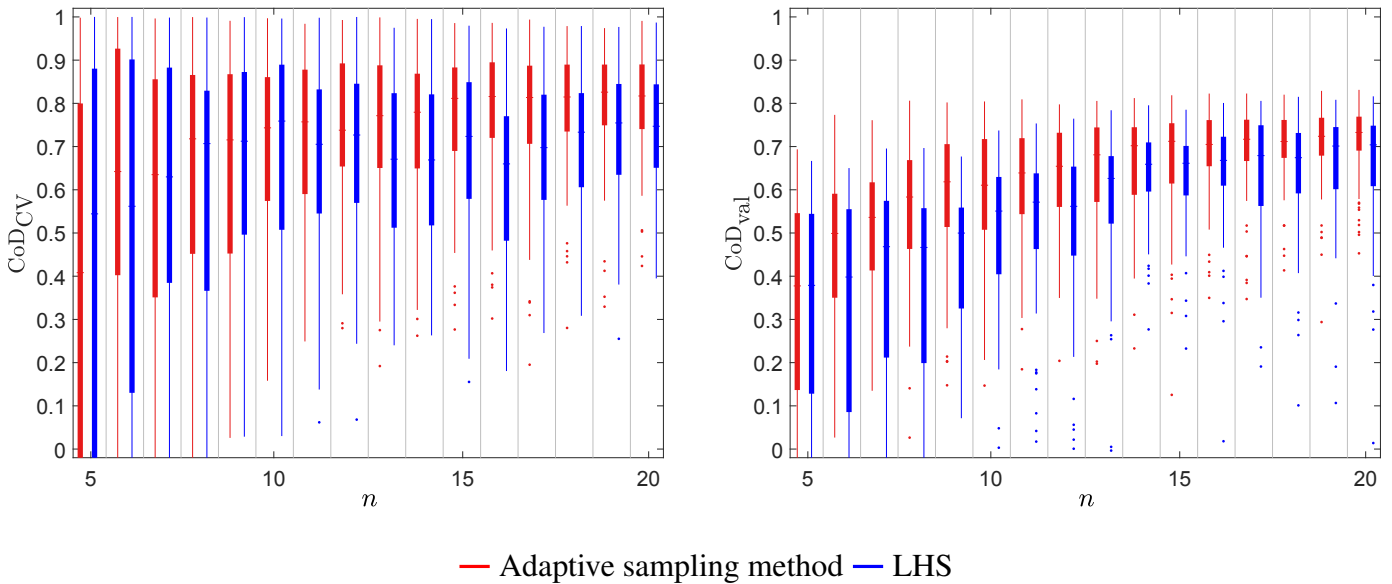

Figure 13: Boxplot of the CoD_{CV} (left) and the CoD_{val} (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of the absolute transfer function corrupted by noise distributed with $\mathcal{N}(0, 0.2)$

approximations even if the data are corrupted by noise. However, if the variance of the overlying noise gets higher, the variation of the noise and of the function cannot be distinguished and the results of both ways of approximation are more and more similar. The reason is that in this case no area is of special interest anymore because of its higher nonlinearity and therefore, new points are sampled over the whole domain. The results of the CoD_{CV} show for $\sigma^2 = 0.1$ again at the beginning (until $n = 10$) a better convergence of the surrogates using LHS training data sets. However, at the latest from $n = 14$ a significant decrease of the variance and improvement of the results is visible for both noise-levels.

4   Computation of sensitivity indices based on the adaptive LS-SVR

With the previous examples it is shown that the distance-based LOO error sampling method mostly improves the quality of the approximation, especially if more points are required in specific areas. However, not only the results of the CoD_{CV} and the CoD_{val} are important, but also it is of interest how the adaptive sampling method affects the calculation of the sensitivity values.

The sensitivity analysis is of importance to determine the most decisive parameters on the response, so that further research could be focused on those parameters first. There are different possibilities to perform a sensitivity analysis such as variance-based methods and derivative-based approaches. In the present work we focus on the application of surrogate models to variance based methods because they have the opportunity of being model-independent and they take the whole input space into account. One often used global variance-based approach is the Sobol indices proposed by Sobol [58], where we observe the first-order effects S_i and the total effects S_{T_i} of each input parameter X_i ($i = 1, ..., n$) on the output Y. The values S_i represent the single influence of each parameter, while the values S_{T_i} contain also the interaction effects between the parameters. They can be calculated using Equation 15 and 16.

$$S_i = \frac{\text{Var}(\mathbb{E}(Y|X_i))}{\text{Var}(Y)} \tag{15}$$

$$S_{T_i} = \frac{\mathbb{E}\left(\text{Var}\left(Y|\mathbf{X}_{\sim i}\right)\right)}{\text{Var}\left(Y\right)} = 1 - \frac{\text{Var}\left(\mathbb{E}\left(Y|\mathbf{X}_{\sim i}\right)\right)}{\text{Var}\left(Y\right)} \tag{16}$$

In these equations $\text{Var}(Y)$ is the unconditional variance and $\mathbb{E}(Y|X_i))$ and $(\mathbb{E}(Y|\mathbf{X}_{\sim i}))$ describe the conditional expected values. Thereby, conditional on $\mathbf{X}_{\sim i}$ means depending on all X_j with $j \neq i$. However, it is usually not possible to calculate these sensitivity values analytical so that the following estimator of the Sobol indices is used [9]:

$$\hat{S}_i^N = \frac{\frac{1}{N}\sum_{j=1}^{N} y_j y_j^{X_i} - \left(\frac{1}{N}\sum_{j=1}^{N} y_j\right)^2}{\frac{1}{N}\sum_{j=1}^{N} y_j^2 - \left(\frac{1}{N}\sum_{j=1}^{N} y_j\right)^2}, \quad i = 1, ..., k \tag{17}$$

22

$$\hat{S}_{T_i}^N = 1 - \frac{\frac{1}{N}\sum\limits_{j=1}^{N} y_j' y_j^{X_i} - \left(\frac{1}{N}\sum\limits_{j=1}^{N} y_j\right)^2}{\frac{1}{N}\sum\limits_{j=1}^{N} y_j^2 - \left(\frac{1}{N}\sum\limits_{j=1}^{N} y_j\right)^2}, \quad i = 1,...,k. \tag{18}$$

There $\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1k} \\ \vdots & & \vdots \\ x_{N1} & \cdots & x_{Nk} \end{bmatrix}$ and $\mathbf{X}' = \begin{bmatrix} x_{11}' & \cdots & x_{1k}' \\ \vdots & & \vdots \\ x_{N1}' & \cdots & x_{Nk}' \end{bmatrix}$ are two different random N-dimensional data sets with the function evaluations $y_j = f([x_{j1},...,x_{jk}])$ and $y_j' = f([x_{j1}',...,x_{jk}'])$. Additionally, k data sets are constructed by exchanging the i'th row of $\mathbf{X}'$ with values from $\mathbf{X}$. Thus, we determine the function responses $y_j^{X_i}$ of $[x_{j1}',...,x_{ji-1}',x_{ji},x_{ji+1}',...,x_{jk}']^{\mathrm{T}}$. That means $(k+2)N$ function evaluations are needed to calculate the estimators of the sensitivity values. In order to get good estimations, N have to be high, which causes high computational time and is the reason for applying surrogate models.

By the application of the surrogate models, we get the approximations $\hat{S}_i^{n,N}$ and $\hat{S}_{T_i}^{n,N}$ of the estimators $\hat{S}_i^N$ and $\hat{S}_i^N$ and therefore estimators for the Sobol indices S_i and S_{T_i}. The approximation quality of $\hat{S}_i^{n,N}$ and $\hat{S}_{T_i}^{n,N}$ depends not only strongly on the number of sample points n and N, but also on the distribution of the sample points. The number N does not decisively influences the computational costs because it only concerns the number of evaluations of the relatively cheap surrogate model. In contrast, it is important to keep n low. Through the application of the adaptive sampling strategy, it is to be achieved that an optimal sampling strategy is found with which the number of required function calls can be reduced.

In the following subsections we observe how the approximation quality of the surrogate-based estimator of the Sobol sensitivity values can be improved by applying the adaptive sampling strategy. Each time one hundred simulations with $N = 100\,000$ for different n are performed and displayed in boxplots.

4.1 Computation of sensitivity indices of $f(\mathbf{x}) = x_1 \sin x_1 + x_2^2 \sin x_2 + x_3^3 \sin x_3$

As an example, we investigate the results of a function where the input parameters have very different influence on the output and therefore different sensitivity values. The observed function is the following:

$$f(\mathbf{x}) = x_1 \sin x_1 + x_2^2 \sin x_2 + x_3^3 \sin x_3 \tag{19}$$

within the interval $[0,15]^3$. The third input parameter x_3 has the highest influence on the output variance with a sensitivity value $S_3 = 0.9938$, while the other two parameter have a sensitivity close to zero ($S_1 = 0.0002$ and $S_2 = 0.0061$). The total effects have same values as S_1, S_2, and S_3 because no interaction effects appear.

From the results of the quality criteria depicted in Figure 14, a high improvement by applying the adaptive sampling method is visible. The convergence of the CoD_{CV} and the CoD_{val} to one is significantly faster in the case where the adaptive sampling method is applied. Also from $n = 125$ the variance of the results is lower.
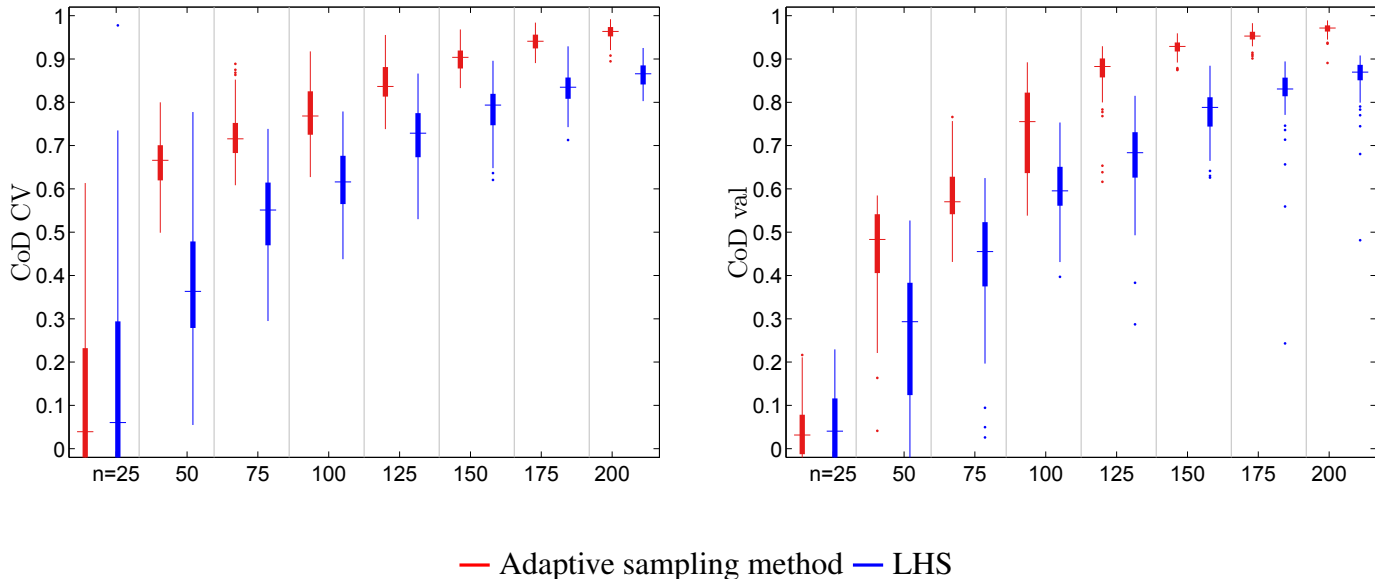
Figure 14: Boxplot of the CoD$_{CV}$ (left) and the CoD$_{val}$ (right) for the distance-based LOO error method (red) and LHS method (blue) depending on n for the approximation of $f(\mathbf{x}) = x_1 \sin x_1 + x_2^2 \sin x_2 + x_3^3 \sin x_3$

Now, we compare the sensitivity values calculated from the different surrogate models with the true sensitivity values. The results of all first-order and total-order effects are shown in Figure 15. Here, we come to the same conclusion, that the adaptive sampling method improves the result visibly. For the two low sensitivity values, the differences between the results of the two methods are visible particularly for the total sensitivity values. It requires more training points to detect a low total sensitivity value if only a LHS data set is used. The results of the third input parameter even show a clear improvement for the first sensitivity value.

4.2 Computation of the sensitivity indices of the model with decaying influence of the variables

In this subsection, we have a look at the example from Subsection 3.4. In the analysis of Expression 14 for $k = 2$ it was visible that the adaptive sampling methods improves the results of the approximation quality at the latest $n = 100$. The results for the sensitivity analysis, shown in Figure 16, provide the same results: The surrogate approximations of the sensitivity values are all closer to the true sensitivity values (dashed line) from $n = 100$ and have a lower variance. Even though, the results do not differ extremely.

For the three-dimensional case, the CoD_{val} indicates a better approximation with the use of the adaptive sampling method from $n = 400$, while the results of the CoD_{CV} advise to use a LHS-data set. Besides, a sufficient approximation quality is not reached with $n = 500$ sample points. This is also visible for the approximation of the sensitivity values plotted in Figure 17 where not always the deviation from the true value for $n = 500$ is acceptable. For all sensitivity

24
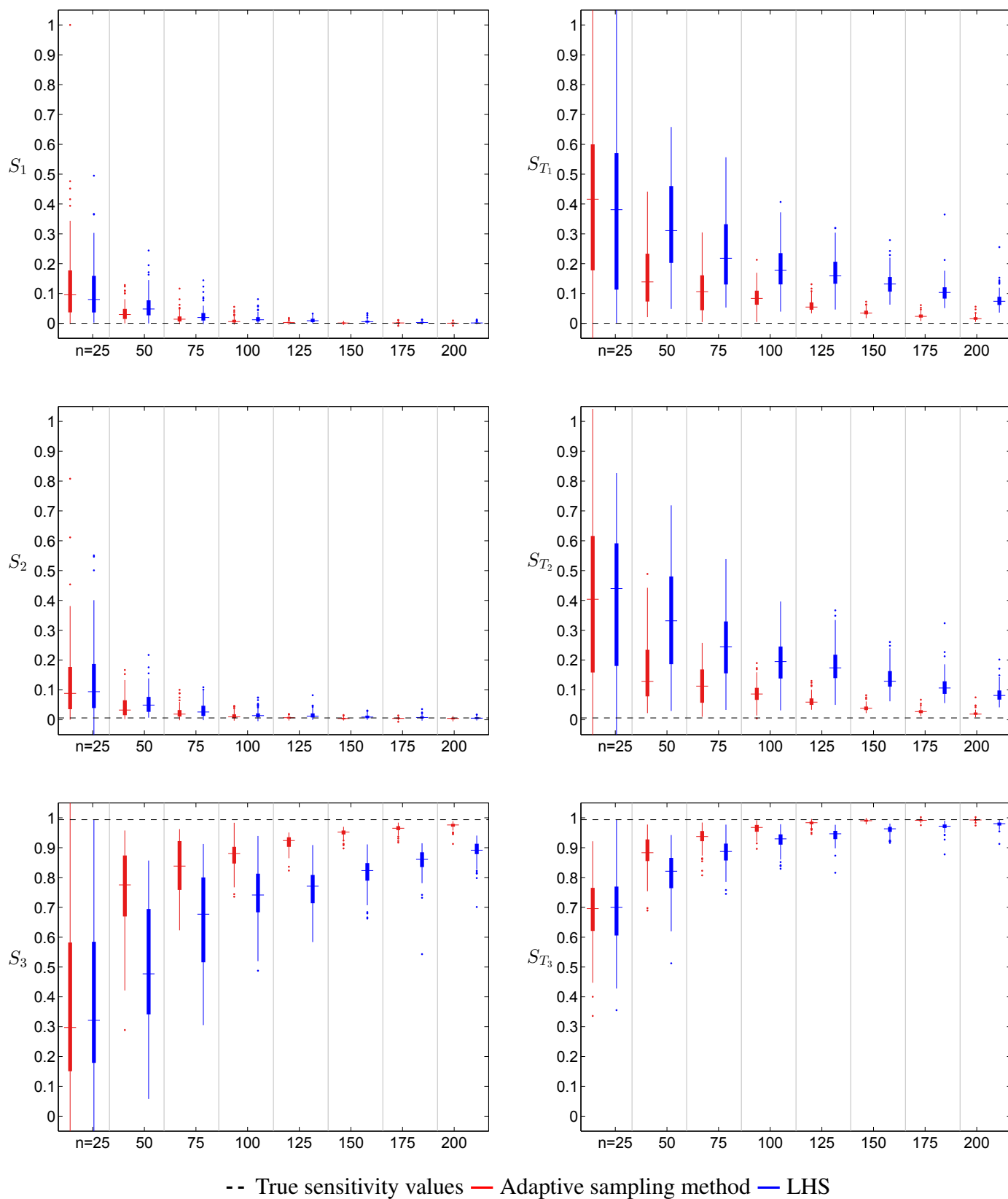
Figure 15: First-order (left) and total-order (right) sensitivity values of $f(\mathbf{x}) = x_1 \sin x_1 + x_2^2 \sin x_2 + x_3^3 \sin x_3$ approximated with the LS-SVR based on the distance-based LOO error method (red) and the LHS method (blue) depending on n
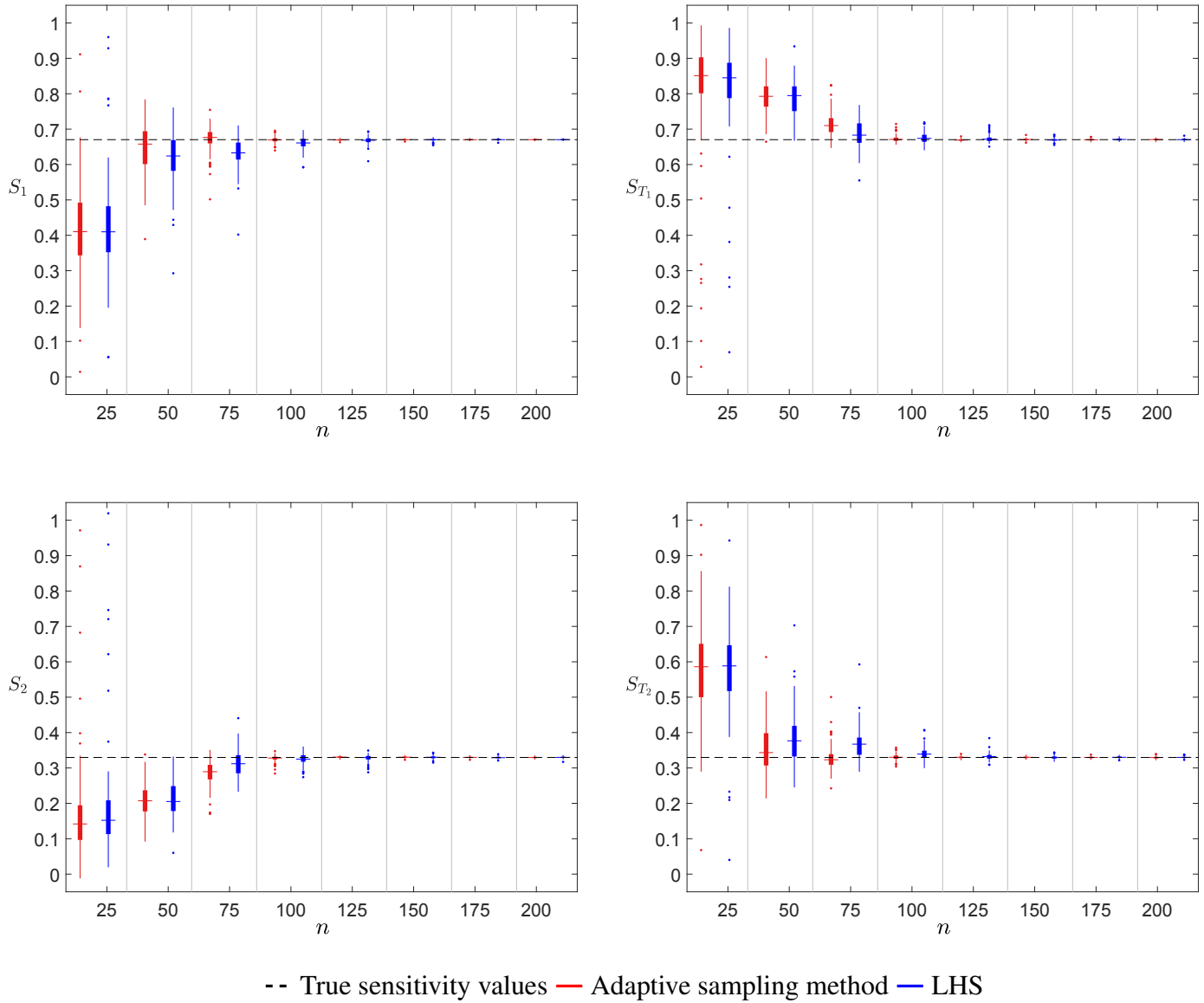
Figure 16: First-order (left) and total-order (right) sensitivity values of $x_1 \sin x_1 + \frac{x_2}{\sqrt{2}} \sin \sqrt{2} x_2$ approximated with the LS-SVR based on the distance-based LOO error method (red) and the LHS method (blue) depending on n

values except S_1, the approximation based on the adaptive sampling method is closer to the true values. Though, sometimes earlier and sometimes later than $n = 400$. An improvement for high n is with the use of the adaptive sampling method definitely visible; however, it is not clearly recognizable from the related quality criteria.

From the results of these two sensitivity examples it can be concluded that the distance-based LOO error method improves the approximation of the true sensitivity values. How great the improvement is and from which n it starts depends on the respective example. It is clear that if the CoD_{CV} advises the use of the distance-based LOO error method, also the sensitivity values based on this method are more accurate. Other way around is not necessarily true as seen in the second example.

5 Closure

In this paper we presented a new global adaptive sampling method and investigate it with different analytical examples. The use of the distance-based LOO error method is of advantage because it identifies areas where the uncertainty of the approximation is the highest; new samples are taken correspondingly. Furthermore, the calculation time remains low because the advantages of the LS-SVR are used. Therefore, it can just be used for the approximation with the LS-SVR, which can be regarded as a disadvantage.

During the numerical analysis, it was visible that the distance-based LOO error method can be used to improve the approximation quality. The method is particularly suitable for functions with critical behavior in some specific regions. Depending on how much the variation of the function changes over the whole region, the global adaptive sampling method is more visibly helpful. Often it makes sense to firstly adapt the global trend of the observed function and later start with the adaptive sampling method to improve the approximation in local regions. One critical point in our investigation is that the CoD_{CV} and the CoD_{val} not always show the same results. In fact, the CoD_{val} is more trustable; however, it cannot be used for real-world-applications because it requires too many sample points. For these problems, we have no other choice than to trust the results of the CoD_{CV}.

By observing the calculation of sensitivity values an improvement by the use of the adaptive method was also visible. This effect was more distinct when the results from the CoD_{CV} match the results from the CoD_{val}. Therefore, the CoD_{CV} is a sufficient criterion for the usefullness of the application of the distance-based LOO error method. In general, it is advised to apply the observed sampling strategy to global sensitivity investigations.

Finally, it should be mentioned that we compared in this paper only the adaptive sampling method with the LHS method because it gives the most suitable space-filling distribution. However, in an engineering application it does not make sense to sample new, larger LHS data sets if the previous number of sample points is not sufficient. In this situation an adaptive sampling method has to be considered, anyway. Therefore, further research will compare the distance-based LOO error method with other global adaptive sampling methods and will apply this method to different engineering examples.
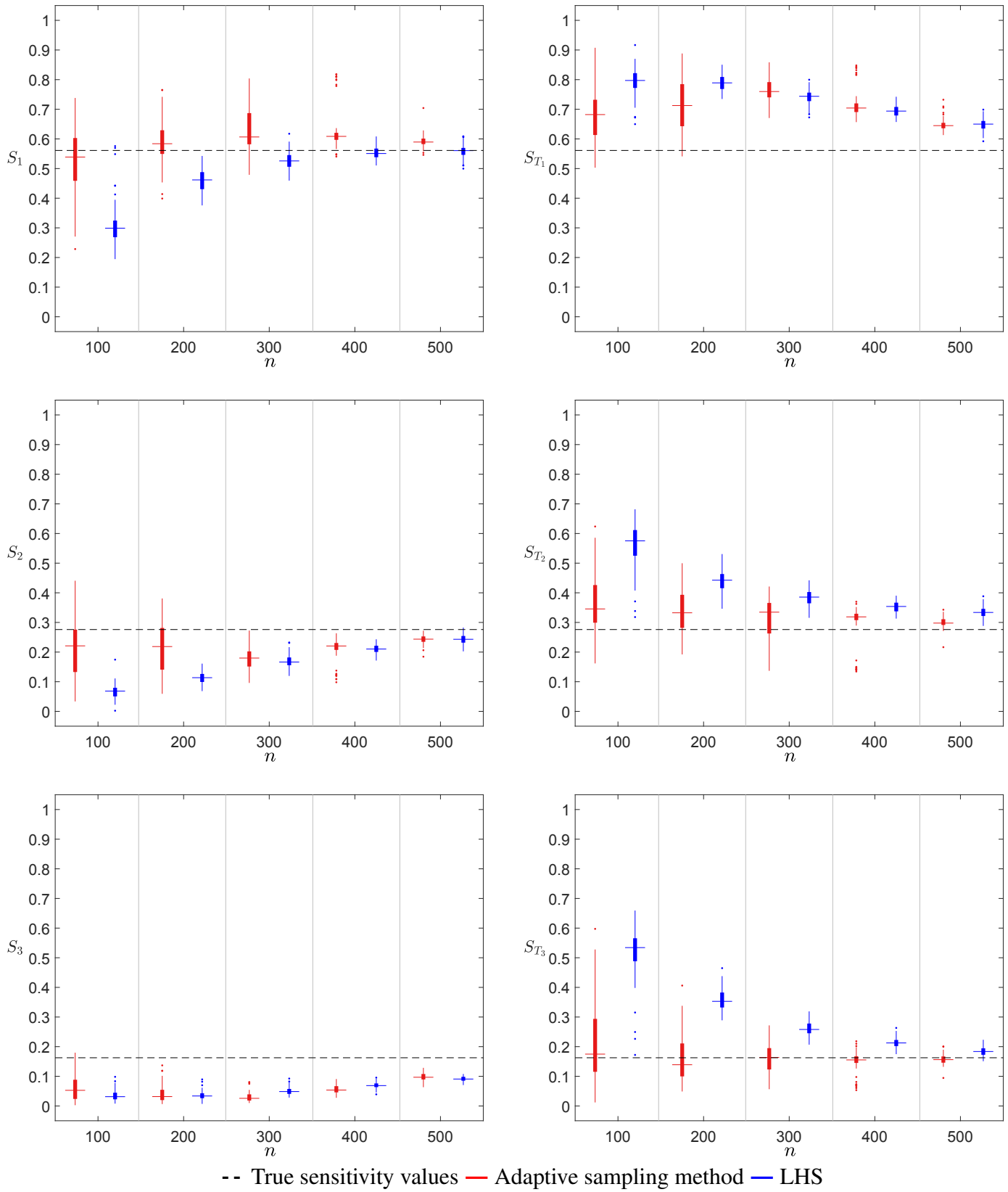
Figure 17: First-order (left) and total-order (right) sensitivity values of $x_1 \sin x_1 + \frac{x_2}{\sqrt{2}} \sin \sqrt{2} x_2 + \frac{x_3}{\sqrt{3}} \sin \sqrt{3} x_3$ approximated with the LS-SVR based on the distance-based LOO error method (red) and the LHS method (blue) depending on n

Acknowledgments

References

[1] W. L. Oberkampf and S. M. DeLand. Error and uncertainty in modeling and simulation. *Reliability Engineering and System Safety*, 75(3):333–357, 2002.

[2] J. C. Helton and D. E. Burmaster. Guest editorial: treatment of aleatory and epistemic uncertainty in performance assessments for complexe systems. *Reliability Engineering and System Safety*, 54(2/3):91–94, 1996.

[3] J. C. Helton, F. J. Davis, and J. D. Johnson. A comparison of uncertainty and sensitivity analysis results obtained with random and Latin hypercube sampling. *Reliability Engineering and System Safety*, 89(3):305–330, 2005.

[4] J. C. Helton, C. W. Hansen, and C. J. Sallaberry. Uncertainty and sensitivity analysis in performance assessment for the proposed high-level radioactive waste repository at Yucca Mountain, Nevada. *Reliability Engineering and System Safety*, 107:44–63, 2012.

[5] J.-T. Hwang, E. P. Dougherty, S. Rabitz, and H. Rabitz. The Green's function method of sensitivity analysis in chemical kinetics. *Journal of Chemical Physics*, 69(11):5180–5191, 1978.

[6] D. G. Cacuci. Sensitivity theory for nonlinear systems. I. Nonlinear functional analysis approach. *Journal of Mathematical Physics*, 22(12):2794–2802, 1981.

[7] L. Vuilleumier, R. A. Harley, and N. J. Brown. First- and second-order sensitivity analysis of a photochemically reactive system (a Green's function approach). *Environmental Science and Technology*, 31(4):1206–1217, 1997.

[8] A. Saltelli and I. M. Sobol. About the use of rank transformation in sensitivity analysis of model output. *Reliability Engineering and System Safety*, 50(3):225–239, 1995.

[9] T. Homma and A. Saltelli. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering and System Safety*, 52(1):1–17, 1996.

[10] M. D. McKay. Nonparametric variance-based methods of assessing uncertainty importance. *Reliability Engineering and System Safety*, 57(3):267–279, 1997.

[11] S. Marzban and T. Lahmer. Conceptual implementation of the variance-based sensitivity analysis for the calculation of the first-order effects. *Journal of Statistical Theory and Practice*, 10(4):589–611, 2016.

[12] A. Saltelli and J. Marivoet. Non-parametric statistics in sensitivity analysis for model output. A comparison of selected techniques. *Reliability Engineering and System Safety*, 28(2):229–253, 1990.

[13] J. C. Helton. Uncertainty and sensitivity analysis techniques for use in performance assessment for radioactive waste disposal. *Reliability Engineering and System Safety*, 42 (2/3):327–367, 1993.

[14] J. P. C. Kleijnen and J. C. Helton. Statistical analyses of scatterplots to identify important factors in large-scale simulations, 1: review and comparison of techniques. *Reliability Engineering and System Safety*, 65(2):147–185, 1999.

[15] J. C. Helton and F. J. Davis. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliability Engineering and System Safety*, 81(1):23–69, 2003.

[16] A. Marrel, B. Iooss, B. Laurent, and O. Roustant. Calculations of Sobol indices for the Gaussian process metamodel. *Reliability Engineering and System Safety*, 94(3):742–751, 2009.

[17] T. H. Andres. Sampling methods and sensitivity analysis for large parameter sets. *Journal of Statistical Computation and Simulation*, 57(1-4):77–110, 1997.

[18] J. P. C. Kleijnen. Sensitivity analysis and related analyses: a review of some statistical techniques. *Journal of Statistical Computation and Simulation*, 57(1-4):111–142, 1997.

[19] J.-M. Bourinet, F. Deheeger, and M. Lemaire. Assessing small failure probabilities by combined subset simulation and support vector machines. *Structural Safety*, 33(6):343–353, 2011.

[20] V. Dubourg, B. Sudret, and J.-M. Bourinet. Reliability-based design optimization using Kriging surrogates and subset simulation. *Structural and Multidisciplinary Optimization*, 44(5):673–690, 2011.

[21] J.-M. Bourinet. Rare-event probability estimation with adaptive support vector regression surrogates. *Reliability Enigneering and System Safety*, 150:210–221, 2016.

[22] A. I. J. Forrester, A. Sobester, and A. J. Keane. *Engineering Design Via Surrogate Modelling: A Practical Guide*. Wiley, 2008.

[23] Z.-H. Han and K.-S. Zhang. *Surrogate-Based Optimization*, chapter 17. InTech Open Access Publisher, 2012.

[24] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering and System Safety*, 93:964–979, 2008.

[25] T. Crestaux, O. Le Maitre, and J.-M. Martinez. Polynomial chaos expansion for sensitivity analysis. *Reliability Engineering and System Safety*, 94(7):1161–1172, 2009.

[26] G. Blatman and B. Sudret. Efficient computation of global sensitivity indices using sparse polynomial chaos expansions. *Reliability Engineering and System Safety*, 95(11):1216–1229, 2010.

[27] M. M. Zuniga, S. Kucherenko, and N. Shah. Metamodelling with independent and dependent inputs. *Computer Physics Communications*, 184(6):1570–1580, 2013.

[28] L. Le Gratiet, C. Cannamela, and B. Iooss. A Bayesian approach for global sensitivity analysis of (multifidelity) computer codes. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):336–363, 2014.

[29] B. Lamoureux, N. Mechbala, and J.-R. Masséb. A combined sensitivity analysis and Kriging surrogate modeling for early validation of health indicators. *Reliability Engineering and System Safety*, 130:12–26, 2014.

[30] B. Iooss and L. Gratiet. Uncertainty and sensitivity analysis of functional risk curves based on Gaussian processes. *Reliability Engineering and System Safety*. doi: 10.1016/j.ress. 2017.11.022.

[31] R.H. Myers. *Response surface methodology*. Allyn and Bacon Inc., 1971.

[32] P. Lancaster and K. Salkauskas. Surfaces generated by moving least squares methods. *Mathematics of Computation*, 37(155):141–158, 1981.

[33] M. D. Buhmann. *Radial Basis Functions*. Cambridge University Press, New York, NY, USA, 2003.

[34] M. J. L. Orr. Regularization in the selection of radial basis function centers. *Neural Computation*, 7(3):606–623, 1995.

[35] D. G. Krige. A statistical approach to some basic mine valuation problems on the Witwatersrand. *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, 52: 119–139, 1951.

[36] H. Drucker, C. J. C Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. *Advances in neural information processing systems 9*, (9):155–161, 1997.

[37] B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.

[38] E. Fiesler and R. Beale. *Handbook of neural computation*. Oxford University Press, Oxford, UK, 1997.

[39] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[40] B. Echard, N. Gaytona, M. Lemaire, and N. Relunb. A combined importance sampling and Kriging reliability method for small failure probabilities with time-demanding numerical models. *Reliability Engineering and System Safety*, 111:232–240, 2013.

[41] B. Gaspar, A. P. Teixeira, and C. Guedes Soares. Adaptive surrogate model with active refinement combining Kriging and a trust region method. *Reliability Engineering and System Safety*, 165:277–291, 2017.

[42] C. Currin, T. Mitchell, M. D. Morris, and D. Ylvisaker. A Bayesian approach to the design and analysis of computer experiments. *Technical Report ORNL – 6498*, 1988.

[43] R. Jin, W. Chen, and A. Sudjianto. On sequential sampling for global metamodeling in engineering design. In *Proceedings of DETC'02 ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 2, 2002.

[44] M. Johnson, L. Moore, and D. Ylvisaker. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, 26:131–148, 1990.

[45] B.E. Boser, I.M. Guyon, and V.N. Vapnik. A training algorithm for optimal margin classifiers. ACM Press, 1992.

[46] C. Cortes and V. Vapnik. Support-vector networks.

[47] V. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, 1995.

[48] J.A.K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9:293–300, 1999.

[49] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. Least squares support vector machines. *World Scientific Pub. Co.*, 2002.

[50] A. Aizerman, E. M. Braverman, and L. I. Rozonoúr. Theoretical foundations of the potential function method in pattern recognition learing. *Automation and Remote Control*, 25 (6):821–837, 1964.

[51] A.J. Smola, P. Bartlett, B. Schölkopf, and D. Schurmans. Advances in large margin classifiers. 2000.

[52] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society Series B*, 36:111–147, 1974.

[53] A. Luntz and V. Brailovsky. On estimation of characters obtained in statistical procedure of recognition. *Technicheskaya Kibernetica*, 3(6), 1969.

[54] G. J. McLachlan, K.-A. Do, and C. Ambrouse. *Analyzing microarray gene expression data*. Wiley, 2004.

[55] G. C. Cawley, N. Talbot, and O. Chapelle. *Estimating predictive variances with kernel ridge regression*, volume 3944, pages 56–77. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2006.

[56] R. M. Neal. Slice sampling. *Annal of Statistics*, 31(3):705–767, 2003.

[57] J. Neter, M. H. Kutner, C. J. Nachtsheim, and W. Wasserman. *Applied Linear Statistical Models*. Irwin, Chicago, 1996.

[58] I. M. Sobol. Sensitivity estimates for nonlinear mathematical models. *Mathematical Modeling and Computing Experiment*, 1(4):407–414, 1993.