Comparative Study of Kriging and Support Vector Regression for Structural Engineering Applications

Maliki Moustapha, Jean-Marc Bourinet, Benoît Guillaume, Bruno Sudret

COMPARATIVE STUDY OF KRIGING AND SUPPORT VECTOR REGRESSION FOR STRUCTURAL ENGINEERING APPLICATIONS

M. Moustapha, J.-M. Bourinet, B. Guillaume, B. Sudret

CHAIR OF RISK, SAFETY AND UNCERTAINTY QUANTIFICATION
STEFANO-FRANSCINI-PLATZ 5
CH-8093 ZÜRICH

Data Sheet

Comparative study of Kriging and support vector regression for structural engineering applications

M. Moustapha[1], J.-M. Bourinet[2], B. Guillaume[3], and B. Sudret[1]

[1] *Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Stefano-Franscini-Platz 5, 8093 Zurich, Switzerland*

[2] *Institut Pascal, CNRS UMR 6602, Sigma Clermont, Campus des Cézeaux, 27 Rue Roche Genès, 63178 Aubière, France*

[3] *Groupe PSA, Route de Gisy, 78943 Vélizy-Villacoublay, France*

Abstract

Metamodeling techniques have been widely used as substitutes of high-fidelity and time-consuming models in various engineering applications. Examples include polynomial chaos expansions, neural networks, Kriging or support vector regression. This papers attempts to compare the latter two in different case studies so as to assess their relative efficiency on simulation-based analyses. Similarities are drawn between these two metamodels types leading to the use of anisotropy for SVR. Such a feature is not commonly used in the SVR related literature. A special care is given to a proper automatic calibration of the model hyperparameters by using an efficient global search algorithm, namely the covariance matrix adaptation - evolution scheme (CMA-ES). Variants of these two metamodels, associated with various kernel or auto-correlation functions, are first compared on analytical functions and then on finite-element-based models. From the comprehensive comparison, it is concluded that anisotropy in the two metamodels clearly improves their accuracy. In general, anisotropic L_2-SVR with the Matérn kernels is shown to be the most effective metamodel.

1   Introduction

Structural design is nowadays routinely performed in order to assess that any newly built structure will operate reliably throughout its lifetime. Such an assessment has been made

mainstream thanks to the advances in computer simulation that allow analysts to reduce their dependence on overly expensive and time-consuming physical testing. Indeed, upon relying on high-fidelity models, one is able to finely predict the behavior of a structure given its operating conditions. However, discrepancies can still be observed when comparing results of model-based simulations and physical experiments. The ubiquitous presence of uncertainty which departs reality from idealized deterministic mathematical modeling is now commonly acknowledged. Methods to characterize uncertainty and to propagate them in a model so as to assess their impact on the model response have been developed in the literature under the framework of *uncertainty quantification* (Sudret, 2007; De Rocquigny, 2012). In structural reliability for instance, the concern is to find the failure probability of a structure given a design configuration. In such analyses, Monte Carlo simulation is the reference method, mostly because it is quite easy and natural to implement. However, its computational burden is too high for complex problems where one can only afford a limited computational budget due to the cost of a single simulation. Variance-reduction techniques, such as importance sampling or subset simulation (Au and Beck, 2001), have been proposed as an alternative in the literature (Asmussen and Glynn, 2007). The improvement they bring is however not sufficient when one needs to simulate from a time-consuming model. Typical examples are transient analyses of structural systems submitted to earthquakes or crash test simulations in the automotive industry where one simulation of a frontal impact may last up to 24 hours on a high-performance computing infrastructure. Such models are unusable in computationally intensive methods like optimization, reliability or sensitivity analysis which are based on simulation methods. Another approach has gained popularity over the past few decades. It consists in considering the model as a black box function for which outputs can only be evaluated on a limited set of inputs and then constructing a simplified mathematical model, called *metamodel* or *surrogate model*, to emulate the underlying mapping of the black box function. Various types of metamodels exist, *e.g.* polynomial response surfaces (Box and Draper, 1986; Kleijnen, 2007), polynomial chaos expansions (Ghanem and Spanos, 2003), artificial neural networks (McCulloch and Pitts, 1988; Broomhead and Lowe, 1988), Gaussian process modeling (Sacks et al., 1989; Santner et al., 2003) or support vector machines (Vapnik, 1998; Smola and Schölkopf, 2004). Methodological developments that combine simulation methods and surrogate modeling have been proposed in the literature to solve complex problems. For instance, polynomial chaos expansions have been used in conjunction with Monte Carlo simulation in Blatman and Sudret (2008, 2010b). Monte Carlo simulation and importance sampling were associated with Kriging in Picheny et al. (2010); Dubourg et al. (2011); Balesdent et al. (2013). Support vector machines were combined with subset simulation in Bourinet et al. (2011); Deheeger and Lemaire (2007) or with importance

sampling in Hurtado (2007). All these applications have shown a dramatic decrease in the number of calls to the original model compared to the use of crude simulation methods, whatever sophisticated the latter may be. The quality of the results however depends on the accuracy of the surrogate models. The first step in implementing such approaches is the choice of the metamodel type. This may depend on the type of problem at hand. Various studies have been carried out to compare different metamodeling approaches with respect to their efficiency and robustness in solving various problems. For instance, Anoop and Messac (2006) compared radial basis functions with Kriging and polynomial response surface. Fang et al. (2005) compared radial basis function and polynomial response surface for multiobjective crashworthiness optimization. Using a variety of metrics, Jin et al. (2000) conducted a benchmark on a wide panel of metamodeling methods which include multivariate adaptive regression splines, radial basis functions, polynomial response surface and Kriging. Zhao and Xue (2010) performed a similar comparative study with almost the same set of metamodels which additionally includes Bayesian neural networks. As can be seen from their publication dates, these comparisons do not take into account the recent development in surrogate modeling. In this contribution, we consider Kriging and support vector regression. Despite their extensive use in the literature, they have been developed and considered by different research communities. To the authors knowledge, only a few studies have considered a cross-comparison between these two metamodels even though they bear some similarities such as the recourse to the kernel or auto-correlation functions for high-dimensional mapping. Besides, each of the methods possesses a number of options whose setting can influence largely the prediction quality. For instance, Li et al. (2010) compared these two metamodels, among others, in the context of decision support systems. The authors argued that a proper choice of the hyperparameters is crucial, yet only a grid search is conducted on a pre-determined candidates set to choose the optimal hyperparameters. Similarly, Nik et al. (2014) performed a comparative study. However, the hyperparameters of the SVR model were simply set according to some heuristics. In this paper, the similarities between Kriging and support vector regression are first highlighted. The selection of the optimal hyper-parameters is introduced in a unified framework for the two metamodels. In this sense, anisotropy, a feature which is widely used in Kriging, is also considered for SVR as suggested in Chapelle et al. (2002). A derandomized stochastic evolution scheme, namely covariance matrix adaptation - evolution strategy (CMA-ES) is then considered for a robust and efficient solution of the resulting high dimensional optimization problem. The paper is organized as follows. In section 1, the basic theory on Kriging and support vector machines is introduced. Section 2 then focuses on how to properly calibrate these two metamodels. The comparative study is finally carried out in Sections 3 and 4 using respectively a set of analytical functions and two finite element

based applications, namely a structural frame and the so-called *sidemember subsystem* under frontal impact.

2 Introduction to metamodeling

2.1 Kriging surrogate

Coming from geostatistics, Kriging has been originally developed by D.G. Krige as a spatial interpolation method (Krige, 1951). It has later been adapted to computer experiments by Sacks et al. (1989).

In this context, let us consider a set of observations: $\mathcal{D} = \{(\boldsymbol{x}^{(i)}, y_i), i \in \{1, \ldots, n\}, \boldsymbol{x}^{(i)} \in \mathbb{X} \subset \mathbb{R}^s, y_i \in \mathbb{Y} \subset \mathbb{R}\}$, where the scalar *outputs* y_i result from an unknown mapping of the s-dimensional *inputs* $\{\boldsymbol{x}^{(i)}, i = 1, \ldots, n\}$. The idea behind Kriging surrogates is to consider the output $y(\boldsymbol{x})$ as a realization of a stochastic process $Y(\boldsymbol{x})$ (Santner et al., 2003):

$$Y(\boldsymbol{x}) = \mu(\boldsymbol{x}) + Z(\boldsymbol{x}), \tag{1}$$

where $\mu(\boldsymbol{x})$ is a deterministic function approximating the mean trend of the output. The departure from this trend is assumed to be a Gaussian process $Z(\boldsymbol{x})$ with zero mean and autocovariance $\text{Cov}\left[Z(\boldsymbol{x}), Z(\boldsymbol{x}')\right] = \sigma^2 R(\boldsymbol{x}, \boldsymbol{x}')$, where σ^2 is the process variance and $R(\boldsymbol{x}, \boldsymbol{x}')$ is the auto-correlation function providing the dependence structure.

The form of the deterministic function and additional hypotheses on the nature of $Y(\boldsymbol{x})$ lead to various types of Kriging (Rasmussen and Williams, 2006; Ginsbourger, 2009). The most general one is *universal Kriging* where the trend is cast as a linear combination of simple regression functions:

$$Y(\boldsymbol{x}) = \sum_{j=1}^{p} \beta_j f_j(\boldsymbol{x}) + Z(\boldsymbol{x}), \tag{2}$$

where $\boldsymbol{\beta} = \{\beta_j, j = 1, \ldots, p\}$ is a weight vector and $\boldsymbol{f} = \{f_j, j = 1 \ldots, p\}$ is a collection of regression functions.

Estimates of the parameters $\boldsymbol{\beta}$ and σ^2 can be derived by maximizing the likelihood function defined from the assumption that the noise $\boldsymbol{Z} = \boldsymbol{Y} - \boldsymbol{F}\boldsymbol{\beta}$ is a correlated Gaussian vector:

$$\begin{aligned}(\widehat{\boldsymbol{\beta}}, \widehat{\sigma^2}) &= \arg\max_{(\boldsymbol{\beta}, \sigma^2) \in \mathbb{R}^p \times \mathbb{R}^{*+}} L(\boldsymbol{\beta}, \sigma^2 | \boldsymbol{y}) = \\ &\frac{1}{((2\pi\sigma^2)^n \det \boldsymbol{R})^2} \exp[-\frac{1}{2\sigma^2}(\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta})^T \boldsymbol{R}^{-1}(\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta})],\end{aligned} \tag{3}$$

where $\boldsymbol{F} = \{F_{ij} = f_j(\boldsymbol{x}^{(i)}), i = 1, \ldots, n, j = 1, \ldots, p\}$.

Solving Eq. (3) provides the estimators $(\widehat{\boldsymbol{\beta}}, \widehat{\sigma^2})$ once $\boldsymbol{R}$ is known:

$$
\begin{aligned}
\widehat{\boldsymbol{\beta}} &= (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F})^{-1} \boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{y}, \\
\widehat{\sigma^2} &= \frac{1}{n} (\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta})^T \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{F}\boldsymbol{\beta}),
\end{aligned}
\tag{4}
$$

where $\boldsymbol{y}$ contains the model response, *i.e.* $\boldsymbol{y} = \{y_1, \ldots, y_n\}^T$.

Eventually, the prediction for a new point is defined by requiring it to respect three conditions, namely linearity with respect to the observed data, unbiasedness and minimal variance. Once the associated problem is solved, one can get the mean and variance formulations of the Universal Kriging estimator (Rasmussen and Williams, 2006):

$$
\begin{aligned}
\mu_{\widehat{Y}}(\boldsymbol{x}) &= \boldsymbol{f}(\boldsymbol{x})^T \widehat{\boldsymbol{\beta}} + \boldsymbol{r}^T \boldsymbol{R}^{-1} (\boldsymbol{y} - \boldsymbol{F}\widehat{\boldsymbol{\beta}}), \\
\sigma_{\widehat{Y}}^2(\boldsymbol{x}) &= \widehat{\sigma^2} (1 - \boldsymbol{r}^T \boldsymbol{R}^{-1} \boldsymbol{r} + \boldsymbol{u}^T (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F})^{-1} \boldsymbol{u}),
\end{aligned}
\tag{5}
$$

where $\boldsymbol{u} = \boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{r} - \boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{r}$ is a vector gathering the auto-correlation function computed between $\boldsymbol{x}$ and each point of the set of observations $\mathcal{D}$:

$$
\boldsymbol{r} = \{r_1, \ldots, r_n\}^T, \quad \text{with:} \quad r_i = R(\boldsymbol{x}, \boldsymbol{x}_i).
\tag{6}
$$

This formulation has many interesting features. First, it is interpolating as shown in Vazquez (2005), meaning that the prediction is exact at the training points and the associated variance is zero. Second it is asymptotically consistent (provided that the auto-correlation function is regular) *i.e.* increasing the size of $\mathcal{D}$ decreases the overall variance of the process. Last, as stated above, the prediction at a given point $\boldsymbol{x}$, is considered as a realization of a Gaussian random variable ($\widehat{Y} \sim \mathcal{N}(\mu_{\widehat{Y}}, \sigma_{\widehat{Y}}^2)$). It is thus possible to derive confidence bounds on a prediction. The variance information is mostly used as an error measure of the epistemic uncertainty of the metamodel due to the sparsity of data. This quantity has favored the fast development of probabilistic *infill sampling criteria* (Jones, 2001; Sasena, 2002) in the context of optimization analyses where the accuracy of the metamodel is improved iteratively only in regions of interest.

2.1.1 The nugget effect

In the case of noisy data, we shall not require the prediction to be interpolating. The commonly-used technique to account for noisy data is the introduction of the so-called *nugget effect* (Matheron, 1971). It consists in making the auto-correlation function discontinuous at the origin.

In practice, an independent white noise $U(x)$ is considered as an additional term of the

stochastic process in Eq. (1):

$$Y(\boldsymbol{x}) = \mu(\boldsymbol{x}) + Z(\boldsymbol{x}) + U(\boldsymbol{x}), \tag{7}$$

where $U(\boldsymbol{x})$ is a zero mean Gaussian variable with variance σ_U^2. This actually corresponds to the addition of a Dirac auto-correlation function. The corresponding auto-covariance is the identity matrix, leading to the following modified expression of the auto-covariance of $Y(\boldsymbol{x})$:

$$\boldsymbol{C} = \sigma^2 \boldsymbol{R} + \sigma_U^2 \boldsymbol{I}_n, \tag{8}$$

where $\boldsymbol{C}$ is the auto-covariance matrix defined by $\{C_{ij} = \mathrm{Cov}\left[\boldsymbol{Z}(\boldsymbol{x}^{(i)}), \boldsymbol{Z}(\boldsymbol{x}^{(j)})\right], i = 1, \ldots, n, j = 1, \ldots, n\}$ and $\boldsymbol{I}_n$ is the identity matrix of size $n \times n$. In practice, the auto-correlation matrix in the previous equations is modified by adding the nugget term at its diagonal.

2.2 Support Vector Regression

2.2.1 Learning machines and structural risk minimization

Support vector machines are a learning machine technique developed by Vapnik and colleagues in the late 1980s. It is based on the principle of *structural risk minimization* where the complexity of the metamodel is controlled to avoid overfitting (Vapnik, 1995, 1998). This is in contrast to *empirical risk minimization* where the best metamodel is selected as the one that minimizes a so-called *empirical risk* (examples are neural networks and radial basis functions). For the empirical risk may only be computed with a finite set of samples, it may potentially differ from the *true risk*. Vapnik (1995) has hence developed a distribution-free bound on the true risk which is a function of the empirical risk and the complexity or capacity of the metamodel. It has been a solid foundation for the development of support vector machines for classification (SVC) and later for regression (SVR) (Smola and Schölkopf, 2004).

2.2.2 SVR formulation

Support vector regression consists in a linear equation that can be cast as follows:

$$f(\boldsymbol{x}, \boldsymbol{w}) = <\boldsymbol{w}, \boldsymbol{x}> + b, \tag{9}$$

where $<, >$ denotes the dot product in $\mathbb{X}$.

To build the model, Cortes and Vapnik (1995) introduced the so-called ε-*insensitive* loss function. Its concept is illustrated in Fig. 1: As long as the difference between the actual and estimated values for a given point is less than ε, the loss function is zero. Beyond this limit, the point is penalized either in a linear (L_1-SVR) or in a quadratic (L_2-SVR) fashion.
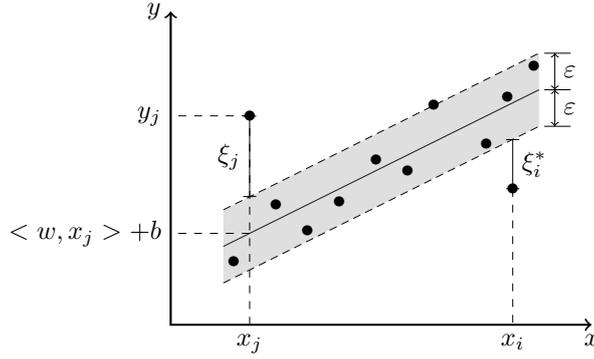
Figure 1: Principle of support vector regression - After Smola and Schölkopf (2004).

For linear penalization, the formulation of the SVR problem reads:

$$\min_{\boldsymbol{w}} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*),$$

$$\text{subject to} \quad y_i - \langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle - b \leq \varepsilon + \xi_i,$$

$$\langle \boldsymbol{w}, \boldsymbol{x}^{(i)}\rangle + b - y_i \leq \varepsilon + \xi_i^*, \tag{10}$$

$$\xi_i, \xi_i^* \geq 0,$$

where ξ_i and ξ_i^* are the so-called *slack variables* which measure the acceptable deviation of a point from the insensitive tube shown as the shaded area in Fig. 1. In this formulation, the minimization of $\frac{1}{2}\|\boldsymbol{w}\|^2$ corresponds to the capacity control. The second term is for empirical risk minimization.

Eq. (10) is a convex quadratic optimization problem with linear constraints and can be solved in its dual formulation by introducing the *Lagrange multipliers*. The Lagrangian of the problem then reads:

$$\begin{aligned}
L(\boldsymbol{w}, b, \boldsymbol{\alpha}) =& \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}(\eta_i\xi_i + \eta_i^*\xi_i^*) \\
& - \sum_{i=1}^{n}\alpha_i(\varepsilon + \xi_i - y_i + \langle \boldsymbol{w}, \boldsymbol{x^{(i)}}\rangle + b) - \sum_{i=1}^{n}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle \boldsymbol{w}, \boldsymbol{x^{(i)}}\rangle - b),
\end{aligned}$$

$$\tag{11}$$

where α_i, α_i^*, η_i and $\eta_i^* \geq 0$ are Lagrange multipliers.

The saddle point of this Lagrangian gives the coefficients $\boldsymbol{\alpha}$ that allow one to make a SVR prediction at any new input point $\boldsymbol{x}$:

$$f(\boldsymbol{x}, \boldsymbol{w}) = <\boldsymbol{w}, \boldsymbol{x}> + b = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) <\boldsymbol{x}^{(i)}, \boldsymbol{x}> + b. \tag{12}$$

This solution has many features. First α_i and α_i^* are the Lagrange multipliers associated

to the constraints on the ε-insensitive tube. This means that when a point $\boldsymbol{x}^{(i)}$ is inside the tube, its associated α_i (or α_i^*) is zero. Besides, α_i and α_i^* cannot be nonzero simultaneously as there is no point $\boldsymbol{x}^{(i)}$ which could lie on the two sides of the tube. Second, the solution only depends on points for which $\alpha_i - \alpha_i^* \neq 0$. Such points are called *support vectors* (SV). It can be shown using the Karush-Kuhn-Tucker optimality conditions of Eq. (10) that $0 \leq \alpha_i^{(*)} \leq C$, where $\alpha_i^{(*)}$ refers to both α_i and α_i^*. The SVs for which $\alpha_i^{(*)} = C$ are called *bounded support vectors* and those for which $0 < \alpha_i^{(*)} < C$ are called *unbounded support vectors*. The former correspond to the points lying strictly outside the ε-insensitive tube, while the latter correspond to those lying at the boundary of the tube defined as the margin. They are useful when it comes to computing the bias term b. In fact, for those SVs, the slack variables vanish and the inequalities in the constraints of Eq. (10) becomes equalities, leading to the following expression for b:

$$ b = \frac{1}{N_{USV}} \sum_{i=1}^{N_{USV}} \left(\mathrm{sign}(\alpha_i - \alpha_i^*)\varepsilon - y_i + \langle \boldsymbol{w}, \boldsymbol{x}^{(i)} \rangle \right), \tag{13} $$

where N_{USV} is the number of unbounded support vectors. Alternatively, when interior-point is used to solve the optimization problem in Eq. (11), b may be simply retrieved as a by-product of the solution (Bompard, 2011).

For L_2-SVR, the quadratic optimization problem to solve for the coefficients $\boldsymbol{\alpha}$ is slightly different (Gunn, 1998). In fact, the penalization term is quadratic thus leading to a different Lagrangian formulation. The solution of this optimization problem gives the coefficients of the expansion which now are no more upper bounded by C.

Many other loss functions such as Huber, Laplace or quadratic and other formulations of SVR exist (Gunn, 1998). In this article, we limit our study to the ε-insensitive loss function and its two formulations (L_1-SVR and L_2-SVR).

2.2.3 Non-linear regression

So far we were only interested in the case of linear regression. There may exist situations where the underlying inputs-outputs mapping is highly non-linear. In such a case, SVR allows non-linear regression. This is achieved by mapping the sampling points into a high-dimensional *feature space* where the transformed data may be handled by linear regression, following the idea by Aizerman et al. (1964). However when doing so, the size of the optimization problem dramatically increases, making it harder to solve. The solution is to resort to the so-called *kernel trick*. Indeed, Eq. (12) shows that the metamodel relies only on the inner product between points in the input space. Consequently, if $\boldsymbol{x}^{(i)}$ is replaced by its image in the feature space $\boldsymbol{\Phi}(\boldsymbol{x}^{(i)})$, the only information needed to construct the metamodel is the

inner product $< \boldsymbol{\Phi}(\boldsymbol{x}^{(i)}), \boldsymbol{\Phi}(\boldsymbol{x}) >$. Thus, there is no need to explicitly know the underlying mapping to the feature space and Eq. (12) becomes:

$$f(\boldsymbol{x}, \boldsymbol{w}) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) k(\boldsymbol{x}^{(i)}, \boldsymbol{x}) + b, \tag{14}$$

where $k(\boldsymbol{x}^{(i)}, \boldsymbol{x})$ represents the inner product in the feature space, usually referred to as the *kernel function*. For a given kernel, the so-called *Mercer's conditions* need to be fulfilled for the underlying mapping to actually exist (Chapelle, 2002; Cherkassky and Mulier, 2007). Examples of well-known kernel functions are polynomials and radial basis functions.

3 Calibration of the two metamodels

3.1 Hyperparameters of the metamodels

3.1.1 Kernel parameters

The generalization ability of these metamodels highly rely on a proper choice of their hyperparameters. Kriging and SVR are both built using respectively the auto-correlation and kernel functions. In the first case, the function relates to the assumption about the nature and the shape of the model being approximated. Rasmussen and Williams (2006) review some commonly-used auto-correlation functions for Kriging. In the second case, the kernel is used to map the data onto a higher dimensional space so as to construct non-linear regression. In practice, similar functions may be used in some cases. A noticeable example is the family of radial basis functions such as the Gaussian auto-correlation/kernel function. In general, for two distinct points x and x', such functions can be expressed in the form $k(x, x', \theta)$ where θ represents the parameter to be tuned. When the problem is multi-dimensional, two possible cases can be assumed:

- Isotropy: In this case, the same parameter is assumed in all dimensions. This may be a sound assumption when the variation rate of the function is similar in all directions. When the input parameters are defined on completely different scales, it is necessary to normalize them so that their ranges become similar in all dimensions.

- Anisotropy: In this case, a different parameter is assumed in each direction. The associated functions may be expressed as follows:

$$k(\boldsymbol{x}, \boldsymbol{x}', \boldsymbol{\theta}) = \prod_{i=1}^{M} k_1(x_i, x_i', \theta_i), \tag{15}$$

where $k_1(\bullet)$ corresponds to the one-dimensional kernel function and θ_i is its parameter

along the i-th dimension. This anisotropy introduces new degrees of freedom to the approximation which results in some cases in higher accuracy. However, when the dimension of the problem is large, the resulting optimization procedure is more difficult to solve, often leading to erroneous models. Henceforth, it is necessary to use appropriate and robust optimization algorithms to calibrate the hyperparameters. The use of anisotropic auto-correlation function is widely acknowledged in Kriging applications. This is not the case for support vector machines where very often only isotropic kernels are considered. Enhanced formulations such as multiple kernels has been introduced (Yeh et al., 2011). Anisotropic kernels were introduced by Chapelle et al. (2002). However, the calibration of these enhanced kernels remain a challenging task. Therefore, they have not been widely used in the literature to the authors knowledge. In this paper, we consider a global optimization scheme, namely the covariance matrix adaptation - evolution scheme (CMA-ES) (Hansen and Kern, 2004) to solve the associated optimization problem.

3.1.2 Specific SVR hyperparameters

Apart from the kernel parameters, SVR has two additional hyperparameters that need to be calibrated, namely the *penalty term C* and the *insensitive tube width ε*. The first parameter introduces some regularization in the minimization of the generalization error by penalizing prediction errors larger than ε as shown in Fig. 1. The higher its value the less errors are tolerated. In contrast, the second parameter allows the model to freely deviate from training points up to a certain value ε. All the points belonging to this insensitive tube are actually not support vectors and do not appear in the expansion in Eq. (14). To the authors experience, by setting the value of C extremely large and that of ε extremely low, the model behaves in prediction similarly to Kriging with a very small nugget, *i.e.* it is almost interpolating. This is the reason why we set very low and high optimization bounds respectively for ε and C in the following applications. To illustrate this idea, let us consider the one-dimensional function $x \mapsto x \sin(x)$ where $x \in [0, 15]$. The results are shown in Fig. 2, where the support vectors are circled in red whereas the insensitive tube is represented by the blue-shaded area. Fig. 2a shows the resulting SVR approximation using optimized values of the hyperparameters. The found solution is actually almost interpolating the data because of the extremely high and low values found respectively for C ($6.38 \cdot 10^{10}$) and ε ($1.80 \cdot 10^{-7}$). In contrast, if we consider values as heuristically set in MATLAB's machine learning toolbox, *i.e.* $C = 19.87$ and $\varepsilon = 0.71$, the produced model is no more interpolating because of the large width of the insensitive tube and the small value of the penalty term (Fig. 2b). Besides, all the training points are no more support vectors as it was the case in the optimized example.
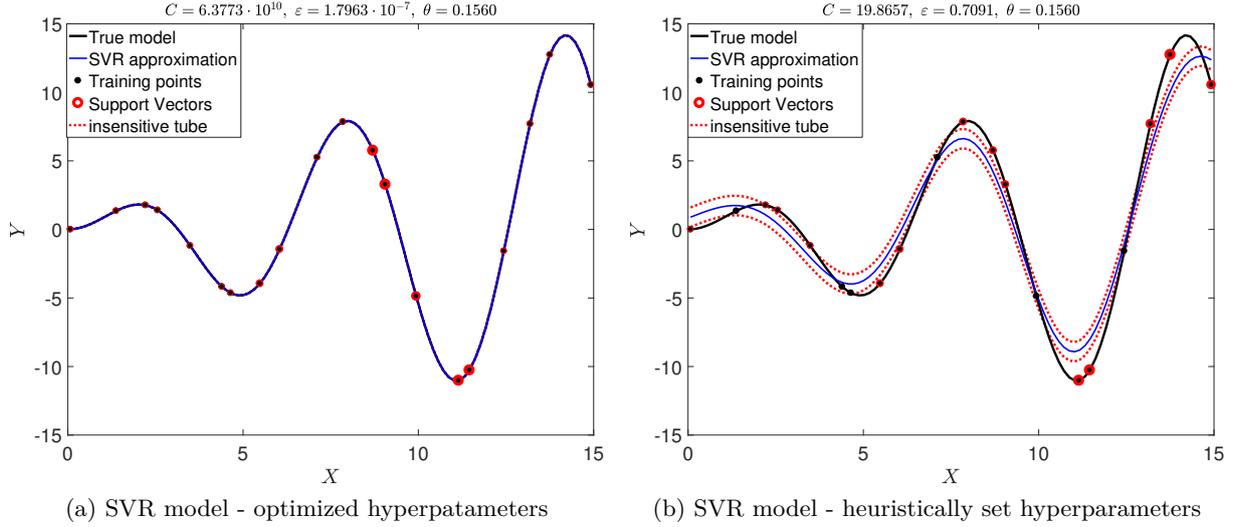
Figure 2: SVR approximation of the function $x \mapsto x\sin(x)$ with optimized hyperparameters and heuristically set values for C and ε.

This simple illustration shows that the optimization of the hyperparameters is a key issue when using SVR, a topic that has been rather overlooked in the recent literature.

Let us now consider the same problem using Kriging. As explained in Section 2.1.1, the nugget can be used either for handling noisy data by requiring the model to not be interpolating or simply to avoid numerical instabilities in the inversion of the auto-correlation matrix. In the latter case, a small value of nugget is often set by default. In the left panel of Fig. 3, a Kriging model is trained using a nugget value of 10^{-10}. The resulting prediction is interpolating as expected, similarly to the SVR case with large C and small ε. However, when we set the nugget to 10^{-1} (Fig. 3b), the resulting model is no more interpolating and the local Kriging variance is never equal to zero. This is due to the large value of the nugget and the resulting behavior is close to that of an SVR with a relatively large ε. In practice, these two parameters can be optimized during the calibration of the model. If the simulation is noisy and if there are enough data points, the resulting nugget values and ε naturally tend to be large. In contrast, with smooth models or when there is not enough data *i.e.* sufficient information for automatic identification of any noise, their values tend to be small. In this paper, all the applications fall in the second category. It should be noted that the Kriging nugget is not optimized in our applications but set by default to 10^{-10}. Furthermore, all input data are normalized before fitting the hyperparameters, as it is common practice.

| $\theta = 0.1560$, nugget $= 10^{-10}$ | $\theta = 0.1560$, nugget $= 10^{-1}$ |
| (a) Kriging model - small nugget ($\sigma_U^2 = 10^{-10}$) | (b) Kriging model - large nugget ($\sigma_U^2 = 10^{-1}$) |
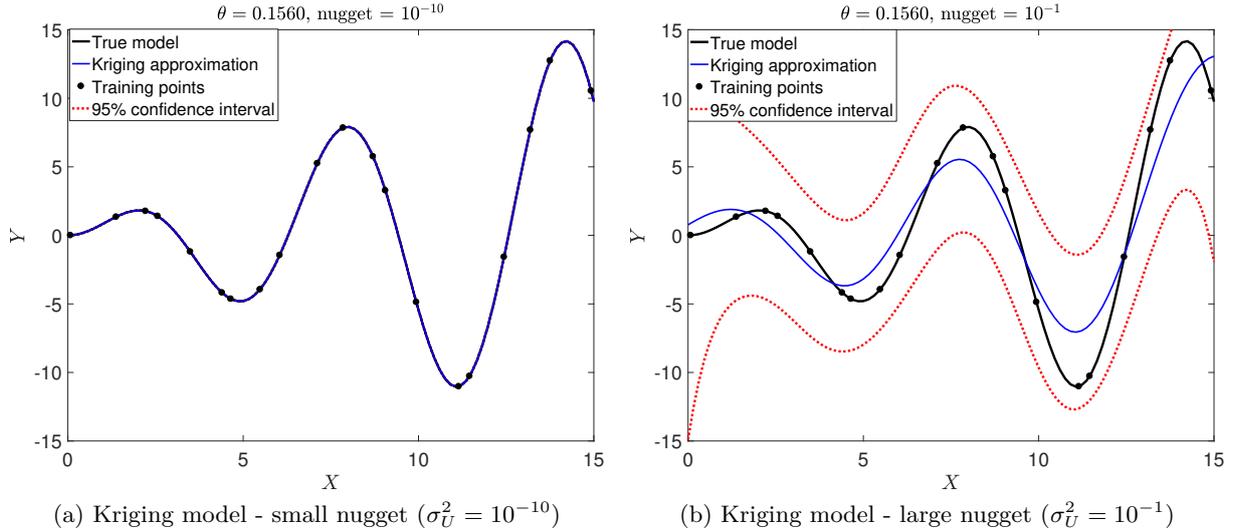
Figure 3: Kriging approximation of the function $x \mapsto x\sin(x)$ with optimized hyperparameters using small and large values of the nugget.

3.2 Model training

3.2.1 Kriging

As said earlier an aspect of prime importance when building a Kriging surrogate is the choice of the auto-correlation function and its parameters. For a given auto-correlation function, there exists many methods for the estimation of its parameters. Among them *variographic analysis* which is mostly used in geostatistics, *cross-validation* where the generalization error is minimized with the available set of observations (Bompard, 2011) or *Bayesian estimation* (O'Hagan, 2006). In computer experiments, the most widely used technique is *maximum likelihood estimation* (Koehler and Owen, 1996; Bachoc, 2013; Dubourg et al., 2011; Roustant et al., 2012; Lophaven et al., 2002a). Gathering the auto-correlation function parameters in $\boldsymbol{\theta}$, it has been shown that the maximum likelihood estimate of $\boldsymbol{\theta}$ is the solution of the following optimization problem:

$$\widehat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^M} \psi(\boldsymbol{\theta}) = \widehat{\sigma}^2(\boldsymbol{\theta}) \det \boldsymbol{R}(\boldsymbol{\theta})^{\frac{1}{n}}, \tag{16}$$

where M is the number of parameters of the auto-correlation function and $\psi(\boldsymbol{\theta})$ is the so-called *reduced likelihood function* (Dubourg, 2011; Marrel et al., 2008).

This reduced likelihood function is not easy to minimize for various reasons. Lophaven et al. (2002a) showed how badly conditioned is the auto-correlation matrix for some values of $\boldsymbol{\theta}$, leading to inaccuracies in the optimization process. $\psi(\boldsymbol{\theta})$ may also have many local minima in the space of the parameters. In the DACE MATLAB toolbox, Lophaven et al. (2002b) use a pattern search algorithm to tackle these problems, and more specifically the *Hook &*

Jeeves method. In the R package *DiceKriging*, Roustant et al. (2012) use a genetic and/or the second-order BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm. In this paper, we consider UQLAB (Marelli and Sudret, 2014), a MATLAB-based framework for uncertainty quantification which offers additional optimization algorithms.

3.2.2 Support vector regression

The generalization ability of a SVR model relies on a good choice of its parameters. They are usually estimated through cross-validation where an error measure is minimized solely on the training points. More specifically, the *leave-one-out* (LOO) error is defined as:

$$e_{loo} = \frac{1}{n} \sum_{p=1}^{n} (f^{\sim p}(\boldsymbol{x}^{(p)}) - y_p)^2, \tag{17}$$

where $f^{\sim p}(\boldsymbol{x}^{(p)})$ is the prediction on the point $\boldsymbol{x}^{(p)}$ by the model built when removing $\boldsymbol{x}^{(p)}$ from the training set.

It is obvious that the actual computation of the LOO error as in Eq. (17), considered in an iterative scheme is unaffordable. In the past few years, many researchers have developed bounds or approximation of the LOO error that are much cheaper to compute. One of them is the *span bound* developed by Chapelle (2002) for classification. Chang and Lin (2005) derived this bound for regression. For L_1-SVR, it reads:

$$\widehat{e}_{loo} = \frac{1}{n} \left(\sum_{p=1}^{n} \left(\alpha_p + \alpha_p^* \right) S_p^2 + \sum_{p=1}^{n} \left(\xi_p + \xi_p^* \right) \right) + \varepsilon \tag{18}$$

The computation of the span bond S_p^2 depends on the nature of the support vector:

- For unbounded support vectors

$$S_p^2 = \frac{1}{\left(\tilde{\boldsymbol{K}}_{USV}^{-1} \right)_{pp}} \quad \text{with} \quad \tilde{\boldsymbol{K}}_{USV} = \left[\begin{array}{cc} \boldsymbol{K}_{USV} & \mathbf{1}_{\text{USV}} \\ \mathbf{1}_{\text{USV}}^T & 0 \end{array} \right]. \tag{19}$$

- For bounded support vectors

$$S_p^2 = k(\boldsymbol{x}^p, \boldsymbol{x}^p) - \widetilde{\boldsymbol{v}}_p^T \boldsymbol{K}_{USV}^{-1} \widetilde{\boldsymbol{v}}_p \tag{20}$$

In these equations, $\boldsymbol{K}_{USV}$ is the matrix of dot products between the unbounded support vectors, $(\tilde{\boldsymbol{K}}_{USV}^{-1})_{pp}$ is the p-th element of the diagonal of matrix $\tilde{\boldsymbol{K}}_{USV}^{-1}$, $\widetilde{\boldsymbol{v}}_p$ is the p-th column of $\boldsymbol{K}_{USV}$ and $\mathbf{1}_{\text{USV}}$ is the identity matrix of size $N_{\text{USV}} \times N_{\text{USV}}$.

For L_2-SVR, the slack variables are no more considered and the LOO estimate reduces

to:

$$\widehat{e}_{loo} = \frac{1}{n}\left(\sum_{p=1}^{n}\left(\alpha_p + \alpha_p^*\right)S_p^2\right) + \varepsilon. \tag{21}$$

The span is now defined by considering the set of all support vectors (Chapelle et al., 2002):

$$S_p^2 = \frac{1}{(\widetilde{\boldsymbol{K}}_{\mathrm{sv}}^{-1})_{ii}}, \tag{22}$$

with $\widetilde{\boldsymbol{K}}_{\mathrm{SV}} = \begin{bmatrix} \boldsymbol{K}_{\mathrm{SV}} & \mathbf{1}_{\mathrm{SV}} \\ \mathbf{1}_{\mathrm{SV}}^T & 0 \end{bmatrix}$ and $\widetilde{\boldsymbol{K}}_{\mathrm{SV}} = [k(\boldsymbol{x}_i, \boldsymbol{x}_j) + 1/C\delta_{ij}]_{i,j\in\mathcal{I}_{\mathrm{sv}}}$. δ_{ij} is the Kronecker symbol, *i.e.* $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ otherwise and $\mathbf{1}_{\mathrm{SV}}$ is the identity matrix of size $N_{\mathrm{SV}} \times N_{\mathrm{SV}}$.

The computation of the LOO error as stated in Eqs. (18) and (21) directly results from the model construction. The only additional operation is the inversion of $\widetilde{\boldsymbol{K}}_{\mathrm{USV}}$ or $\widetilde{\boldsymbol{K}}_{\mathrm{sv}}$. However, this estimate is not a continuous function of the model parameters. Vapnik and Chapelle (2000) smoothened it by introducing a regularization term in the expression of S_p^2. This allowed them to consider gradient methods to find the optimal parameters of the model. In this paper, we use a stochastic global optimization technique to minimize $\widehat{e}_{loo}$ and we find it is not necessary to smoothen the estimates for proper optimization.

3.3 Hyperparameters calibration using hybrid CMA-ES

The Covariance Matrix Adaptation - Evolution Strategy (CMA-ES) is a derandomized stochastic search algorithm introduced by (Hansen and Ostermeier, 2001). It relies on an evolution strategy which appropriately adapts the covariance matrix of a normal distribution by increasing its variance along the directions that have increased the fitness in the recent past iterations. The updating scheme also relies on information from older iterations which is set to decay slowly.

In the so-called (μ, λ)-CMA-ES strategy, the candidate solutions of the next generation are sampled using the normal distribution as follows (Hansen and Kern, 2004; Hansen, 2005):

$$\boldsymbol{m}_k^{(g+1)} \sim \langle\boldsymbol{m}^{(g)}\rangle + \sigma^{(g)}\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{(g)}), \qquad \text{for } k = 1, \ldots, \lambda, \tag{23}$$

where $\boldsymbol{m}_k^{(g+1)}$ is the k-th offspring at generation $g+1$, $\sigma^{(g)}$ is the global step size and $\boldsymbol{\Sigma}^{(g)}$ is the covariance matrix at generation g. The so-called *recombination point* $\langle\boldsymbol{m}^{(g)}\rangle$ is obtained as the weighted means of μ selected individuals, *i.e.*

$$\langle\boldsymbol{m}^{(g)}\rangle = \sum_{i=1}^{\mu} w_i \boldsymbol{m}_{i:\lambda}^{(g)} \quad \text{for } i = 1, \ldots, \mu \tag{24}$$

where $w_i > 0, i = 1, \ldots, \mu$ are weighting coefficients and $\boldsymbol{x}_{i:\lambda}^{(g)}$ denotes the i-th best solution from generation g.

The covariance matrix is adapted using an exponentially fading record of all successful previous steps. This is practically implemented by first monitoring a so-called *evolution path* which reads:

$$\boldsymbol{p}_c^{(g+1)} = (1 - c_c)\boldsymbol{p}^{(g)} + \sqrt{c_\sigma(2 - c_\sigma)}\boldsymbol{B}^{(g)}\boldsymbol{D}^{(g)}\boldsymbol{B}^{(g)}\frac{\sqrt{\mu_{eff}}}{\sigma^{(g)}}(\langle \boldsymbol{m}^{(g+1)} \rangle - \langle \boldsymbol{m}^{(g)} \rangle), \qquad (25)$$

where $\mu_{eff} = 1/\sum_{i=1}^{\mu} w_i^2$ is the so-called *variance effective selection mass* and the matrices $\boldsymbol{B}^{(g)}$ and $\boldsymbol{D}^{(g)}$ are obtained using a principal component analysis of $\boldsymbol{\Sigma}^{(g)}$, *i.e.* $\boldsymbol{\Sigma}^{(g)} = \boldsymbol{B}^{(g)}\boldsymbol{D}^{(g)^2}\boldsymbol{B}^{(g)^T}$. The global step size is updated using the following rule:

$$\sigma^{(g+1)} \approx \sigma^{(g)}\exp(\frac{c_\sigma}{d_\sigma}(\frac{\|\boldsymbol{p}_\sigma^{(g+1)}\|}{\sqrt{s}(1 - \frac{1}{4s} + \frac{1}{21s^2})} - 1)), \qquad (26)$$

where s is the dimension of the optimization problem. The remaining parameters in these two equations (*i.e.* c_c, c_σ, d_σ) are coefficients whose optimal values can be found in Hansen and Kern (2004).

Once the evolution path is updated, the covariance matrix can be adapted as follows:

$$\begin{aligned}
\boldsymbol{\Sigma}^{(g+1)} = (1 - c_{cov})\boldsymbol{\Sigma}^{(g)} + \frac{c_{cov}}{\mu_{cov}}\boldsymbol{p}_c^{(g+1)}(\boldsymbol{p}_c^{(g+1)})^T \\
+ c_{cov}(1 - \frac{1}{\mu_{cov}})\sum_{i=1}^{\mu}\frac{w_i}{\sigma^{(g)^2}}(\boldsymbol{m}_{i:\lambda}^{(g+1)} - \langle \boldsymbol{m}^{(g)} \rangle)(\boldsymbol{m}_{i:\lambda}^{(g+1)} - \langle \boldsymbol{m}^{(g)} \rangle)^T,
\end{aligned} \qquad (27)$$

where c_{cov} and μ_{cov} are CMA-ES internal parameters whose values can be found in Hansen and Kern (2004).

It should be noted here that a degenerate case of CMA-ES is the cross-entropy method for optimization which has been successfully used by Bourinet (2015) for the calibration of support vector machines classification's hyperparameters in high dimension. Here we consider directly CMA-ES in its general formulation as it has shown to be an efficient and robust search algorithm, especially for ill-posed problems (Hansen and Kern, 2004). The suggested optimal population size is usually given as $\lambda = 4 + \lfloor 3\ln(s) \rfloor$ and $\mu = \lfloor \lambda/2 \rfloor$ where $\lfloor \bullet \rfloor$ denotes the floor function. These optimal values, which actually grow sub-linearly with respect to the dimension s allow, CMA-ES to be an extremely efficient local search algorithm as it was originally intended. However, by increasing the population size, one achieves also an efficient *global* search algorithm at has been shown by various benchmark problems in Hansen and Kern (2004). In this paper, we consider $\lambda = 20(4 + \lfloor 3\ln(s) \rfloor)$ and $\mu = \lfloor \lambda/2 \rfloor$.

4 Comparative study

4.1 Description of the study

Kriging and SVR models are built in the different configurations introduced above *i.e.* using isotropic and anisotropic kernels and with L_1- and L_2-penalization for SVR. UQLAB, a MATLAB-based framework for uncertainty quantification is used in this purpose (Marelli and Sudret, 2014). The span estimate of the leave-one-out error is computed as described above following Chapelle et al. (2002) and Chang and Lin (2005) for the calibration of the SVR model. Maximum likelihood estimation is considered for the calibration of the Kriging models. The resulting optimization problems are solved using hybrid CMA-ES, *i.e.* a CMA-ES optimization procedure whose solution is refined by a gradient-based approach.

Three different kernel functions are considered which read in their isotropic form as follows:

- Gaussian:
$$k(\boldsymbol{x}, \boldsymbol{x}'; \theta) = \exp(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{2\theta^2}); \tag{28}$$

- Matérn 3/2:
$$k(\boldsymbol{x}, \boldsymbol{x}'; \theta) = (1 + \sqrt{3}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\theta}) \exp{(-\sqrt{3}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\theta})}; \tag{29}$$

- Matérn 5/2:

$$k(\boldsymbol{x}, \boldsymbol{x}'; \theta) = (1 + \sqrt{5}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{l} + \frac{5}{3}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{\theta^2}) \exp{(-\sqrt{5}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\theta})}. \tag{30}$$

In these three equations, θ stands for the characteristic length-scale.

The metamodels are compared with respect to three error metrics, namely:

- *the normalized mean-square error*:

$$NMSE = \frac{\sum_{i=1}^{n_{test}}(f(\boldsymbol{x}^{(i)}) - y_i)^2}{\sum_{i=1}^{n_{test}}(\bar{y} - y_i)^2}, \tag{31}$$

- *the normalized average absolute error:*

$$NAAE = \frac{\sum_{i=1}^{n_{test}}|f(\boldsymbol{x}^{(i)}) - y_i|}{n_{test}\,\sigma_Y}, \tag{32}$$

- and *the normalized maximum absolute error:*

$$NMAE = \frac{\max_{i=1}^{n_{test}}|f(\boldsymbol{x}^{(i)}) - y_i|}{n_{test}\,\sigma_Y}, \tag{33}$$

where n_{test} is the size of the validation set, $f(\boldsymbol{x}^{(i)})$ and y_i are respectively the estimated

and actual outputs at the point $\boldsymbol{x}^{(i)}$, $\bar{y}$ is the mean value of the actual outputs in the validation set and $\sigma_Y = \sqrt{1/(n_{test} - 1) \sum_{i=1}^{n_{test}} (\bar{y} - y_i)^2}$. The first two error metrics (NMSE and NAAE) give the metamodel's global accuracy measure while the third one rather relates to a local measure of accuracy.

Three analytical functions of different dimensions are used to generate the training and validation sets. The training samples are generated on a hypercube following three different designs of experiments:

- Sobol' quasi-random numbers, as originally developed in Sobol (1967);

- Centroidal Voronoi Tessellations (CVT) which are geometrically optimized initial designs based on the so-called *Voronoi regions* (Du et al., 1999);

- Optimized latin hypercube (OLH) which here corresponds to an initial Latin Hypercube design (McKay et al., 1979) optimized with respect to its L_2-discrepancy as proposed in Franco (2008).

The different experimental design types are introduced here, not as a mean to assess their respective efficiency per se, but to increase the robustness of the computed error metrics through replications. In this sense, we only present the median values of the resulting errors for each configuration. This allows us to reduce the number of printed comparative figures in the same time.

4.1.1 Two-dimensional function

For this first example, we compare the metamodels on a two-dimensional function, which reads:

$$
\begin{aligned}
f(\boldsymbol{x}) = {} & 0.75 \exp\left(-\frac{(9x_1 - 2)^2}{4} - \frac{(9x_2 - 2)^2}{4}\right) + 0.75 \exp\left(-\frac{(9x_1 + 1)^2}{49} - \frac{9x_2 + 1}{10}\right) \\
& + 0.5 \exp\left(-\frac{(9x_1 - 7)^2}{4} - \frac{(9x_2 - 3)^2}{4}\right) - 0.2 \exp\left(-(9x_1 - 4)^2 - (9x_2 - 7)^2\right),
\end{aligned}
\tag{34}
$$

where the input $\boldsymbol{x} = \{x_1, x_2\}^T \in [0, 1]^2$.

This function is highly non-linear and features two Gaussian peaks and a local minimum as illustrated in Fig. 4. For each design, three sizes are considered: 10, 20 and 40. The validation set consists of a Monte-Carlo sample set of 10^6 points.

The median global error metrics (NMSE and NAAE) computed with respect to each experimental design type are shown in Fig. 5. For $n = 10$ and $n = 20$, all the metamodels exhibit similar errors, even though Kriging performs in general slightly better than SVR. The latter shows better results when using the Matérn kernels instead of the Gaussian ones. Finally, for $n = 40$, the error are dramatically reduced as expected since the design space
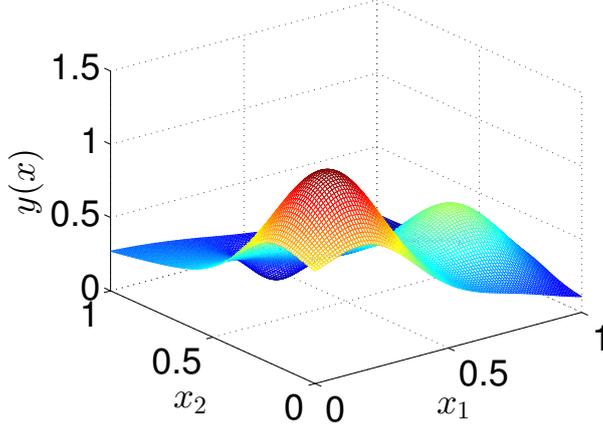
Figure 4: Representation of the two-dimensional function.

is densely sampled. However, there are some abnormally large errors produced by SVR, especially with the Gaussian kernel. This is due to difficulties encountered while optimizing the hyperparameters. In the case of the Gaussian kernel, for some reasons, matrices $\boldsymbol{K}_{USV}$ and $\boldsymbol{K}_{sv}$ in Eqs. (19),(20),(22) are not well conditioned when considering some values of the hyperparameters. This makes their inversion difficult and the overall optimization problem extremely noisy and thus difficult to solve.

The normalized maximum absolute errors are shown in Fig. 6. The trend is the same as for the global error except that the difference between the two metamodels is not so large in terms of relative errors.

For all the metrics (global and local), anisotropy does not seem to improve the results substantially in this example. This may be explained by the shape of the function as shown in Fig. 4. The characteristic length-scale should be roughly the same for each direction. While using anisotropy, the optimization problem size is thus increased uselessly.

As a final remark, it can be pointed out that NMSE and NAAE exhibit the same trends. This is also observed for all the examples considered in this paper as should expected given Eqs. (31) and (32). In the sequel, we thus present only the NMSE and NMAE results.

4.1.2 Critical Euler force limit-state function

The second example is related to the buckling of a simply supported beam of length L with a rectangular cross-section $b \times h$ $(h > b)$ submitted to a normal compressive load F_{ser}. The Euler critical load for buckling is $F_{cr} = \frac{\pi^2 EI}{L^2}$ where $I = \frac{bh^3}{12}$. The margin-to-buckling is then obtained by:

$$y(\boldsymbol{x}) = \frac{\pi^2 E b h^3}{12 L^2} - F_{ser}, \qquad (35)$$

where $\boldsymbol{x} = \{E, b, h, l, F_{ser}\}^T$. The variation ranges of the parameters are given in Table 1.

The results for the three sizes of experimental designs (25, 50 and 100) are gathered in

18
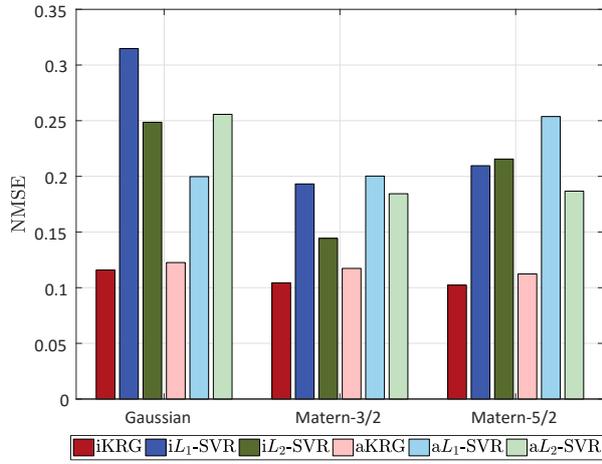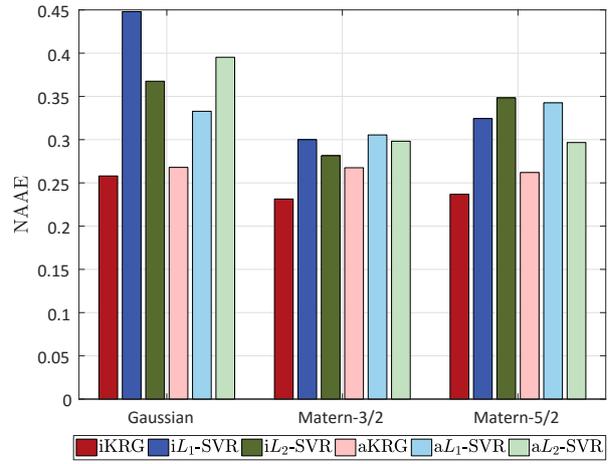
(a) $NMSE$ with $n = 10$

(b) $NAAE$ with $n = 10$
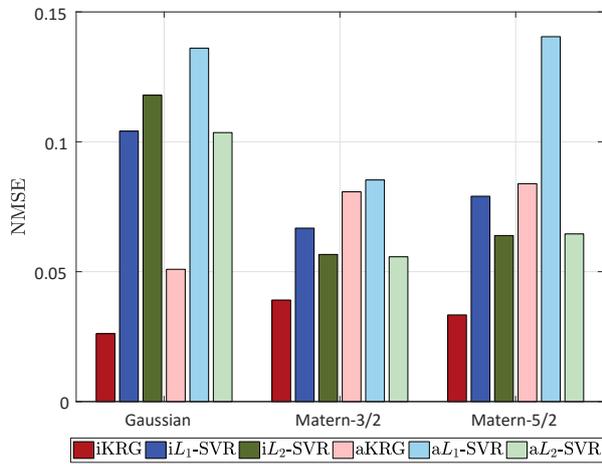
(c) $NMSE$ with $n = 20$
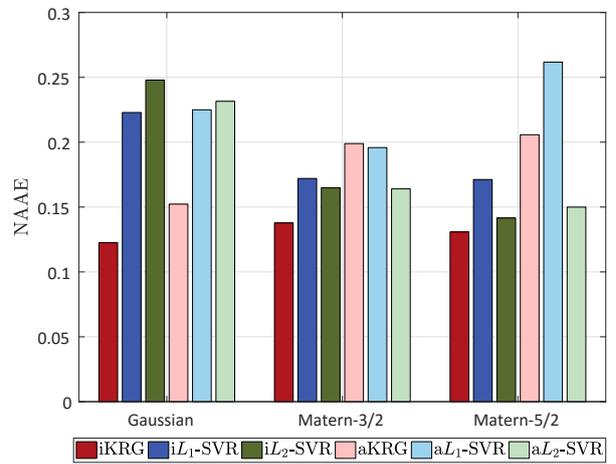
(d) $NAAE$ with $n = 20$
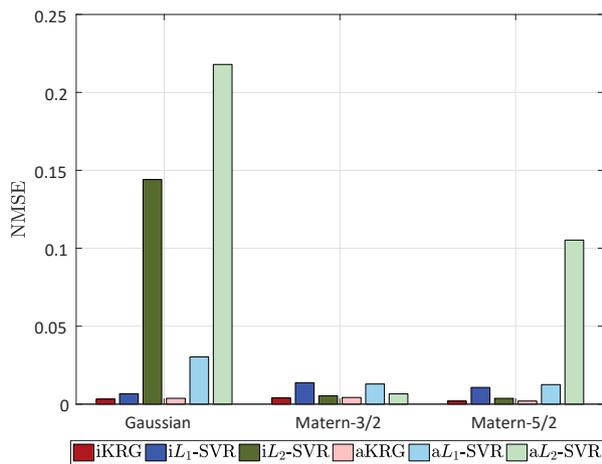
(e) $NMSE$ with $n = 40$
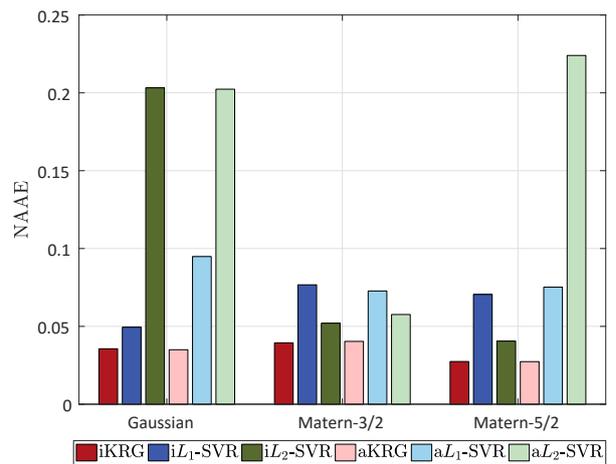
(f) $NAAE$ with $n = 40$

Figure 5: Two-dimensional analytical function - Median values of the normalized mean square and normalized average absolute errors.
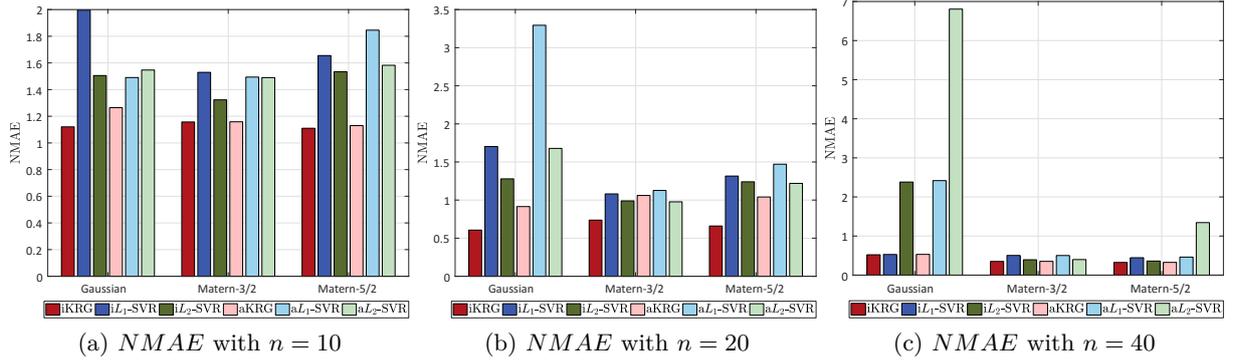
Figure 6: Two-dimensional analytical function - Median values of the normalized maximum absolute error.

Table 1: Inputs parameters for the critical Euler force limit state function.

Parameter	Young's modulus	width	height	length	Service load
	E (MPa)	b (m)	h (m)	L (m)	F_{ser} (kN)
Minimum	60000	0.10	0.25	1	10
Maximum	80000	0.25	0.30	3	70

Fig. 7. Contrary to the previous example, anisotropy plays an important role here as the errors are dramatically decreased when considering anisotropic kernels. Whether focusing only on isotropic cases or not, SVR is slightly better than Kriging on this example. In some cases, such as with the Matérn 3/2 kernel, isotropic L_2-SVR even leads to slightly better results than anisotropic Kriging with respect to the normalized maximum absolute error.

4.1.3 Sobol' g-function

For this last example, we consider the Sobol' g-function. It is a function widely used in global sensitivity analysis as its sensitivity indices can be expressed analytically (Marrel et al., 2008). We use it here for benchmarking surrogate models since it is fairly complex and its parameters can be tuned to control the importance of a given dimension. The function reads:

$$y(\boldsymbol{x}) = \prod_{i=1}^{M} \frac{|4x_i - 2| + a_i}{1 + a_i}, \tag{36}$$

where $M = 20$ is the dimension of the problem and a_i are coefficients which control the importance of a variable. As indicated in Marrel et al. (2008), it can roughly be said that the dimension i is important when $a_i \to 0$ and becomes insignificant for $a_i \geq 100$. In this paper, we choose to have some non-important variables as it is likely to happen in real physical high-dimensional problems. Besides, we alternate the level of importance between
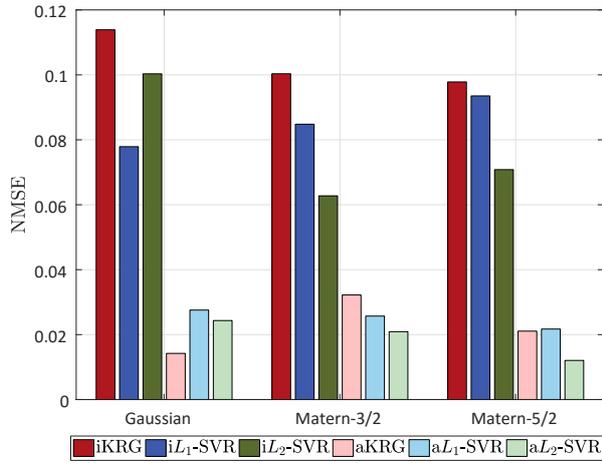
(a) $NMSE$ with $n = 25$
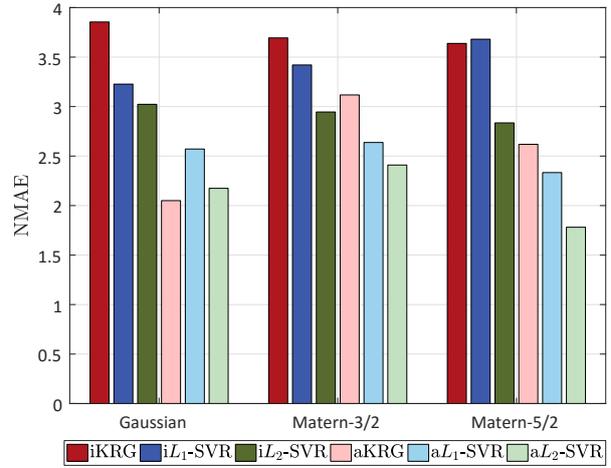
(b) $NMAE$ with $n = 25$
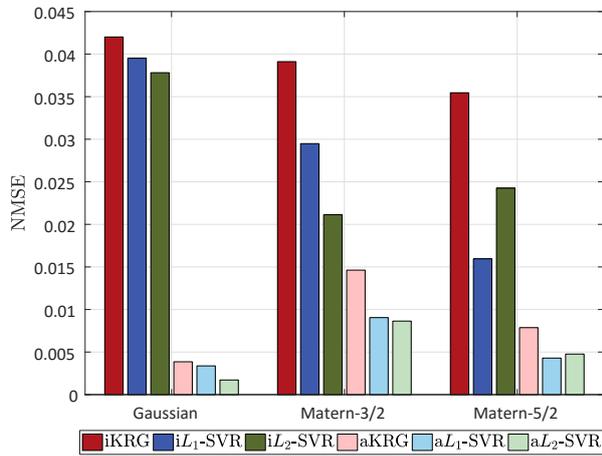
(c) $NMSE$ with $n = 50$
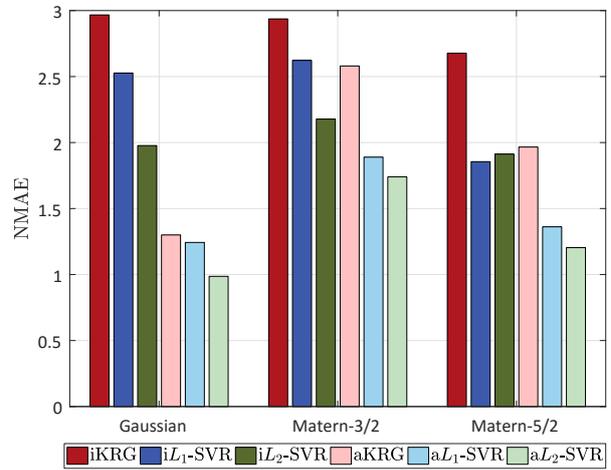
(d) $NMAE$ with $n = 50$
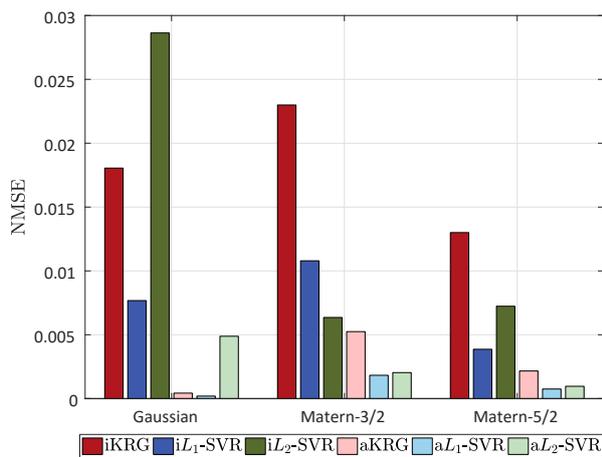
(e) $NMSE$ with $n = 100$
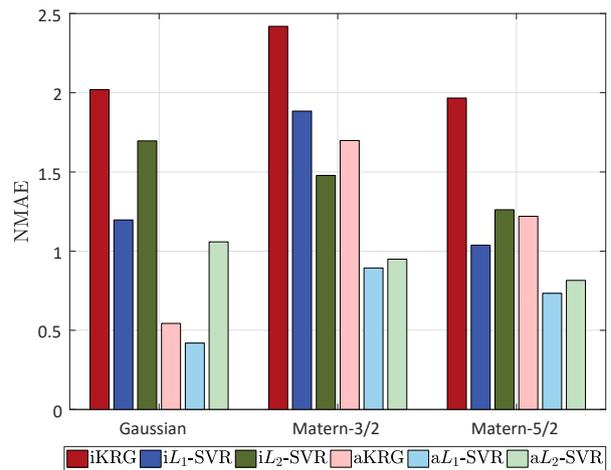
(f) $NMAE$ with $n = 100$

Figure 7: Euler critical force limit-state function - Median values of the normalized mean square and normalized maximum absolute errors.

the dimensions so that we have:

$$a_i = \begin{cases} 3(i-1) & \text{if } i \text{ is odd,} \\ 30i & \text{if } i \text{ is even.} \end{cases} \qquad (37)$$

The validation set is reduced to a Monte Carlo of 10^5 points. Fig. 8 shows the different results. Once again, as expected anisotropy increases the metamodels accuracy in all cases. In general, isotropic models are not able to approximate this high-dimensional highly anisotropic Sobol function accurately, except for L_2-SVR with the Matérn 5/2 kernel. Same trends are observed with NMSE and NMAE. Whether considering isotropic or anisotropic cases, the increase in the experimental design size does not considerably improves the metamodels accuracy. Given the dimensionality of the problem ($M = 20$), such increases hardly modifies the data point density in the space *i.e.* the experimental design remains scarce.

5 Applications using finite element models

5.1 Frame structure

The first example we consider in this section is a three-span five story frame structure as sketched in Fig. 9. This structure has been used in structural reliability for various applications. It was considered by Liu and Der Kiureghian (1991) and by Wei and Rahman (2007), respectively for assessing optimization algorithms for reliability analysis and for benchmarking various reliability methods. It was more recently considered in Blatman and Sudret (2010a) for illustrating the efficiency of sparse polynomial chaos expansions. The frame structure is horizontally loaded and the quantity of interest is the horizontal displacement in the right corner of the top floor. The model consists of 21 parameters gathering the three lateral loads, eight moments of inertia and cross-sectional areas and two Young's moduli. In the original problem, these parameters follow lognormal and truncated Gaussian distributions. The design space considered to build the SVR and Kriging models is defined by considering a mean-centered bounded area covering 95% of the probability mass in each dimension. The lower and upper bounds are therefore set using respectively the $0.025-$ and $0.975-$percentiles computed over a large Monte Carlo set (10^6 points). The experimental design consists of a Latin hypercube sampling of size 214, which is the optimal size obtained for sparse PCE in Blatman and Sudret (2010a). The results are shown in Fig. 10. Once again anisotropic L_1-SVR with the Gaussian kernel does not appropriately approximate the function leading to an extremely large error both in terms of $NMSE$ and $NMAE$. The other metamodels produce extremely high accurate approximations as in Blatman and Sudret (2010a). Anisotropy slightly improves the prediction accuracy. A closer look on the
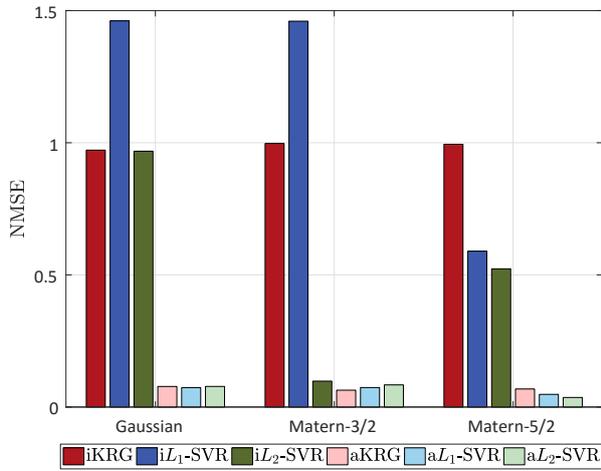
(a) $NMSE$ with $n = 100$
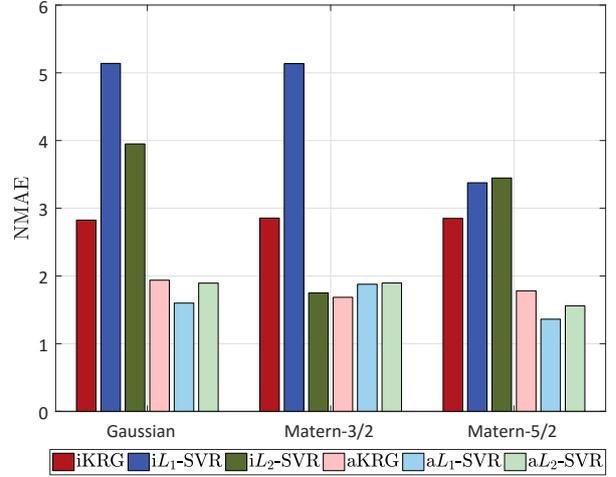
(b) $NMAE$ with $n = 100$
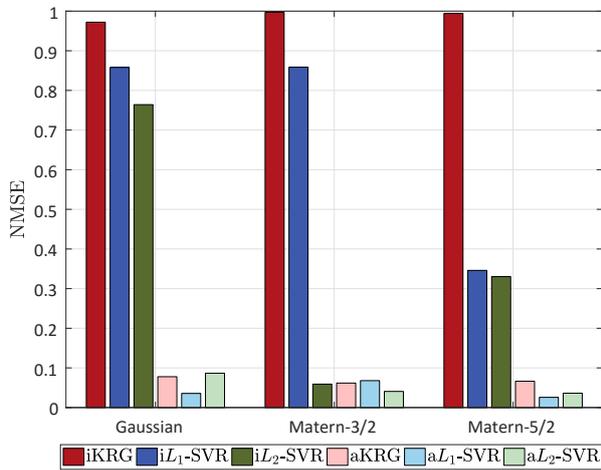
(c) $NMSE$ with $n = 200$
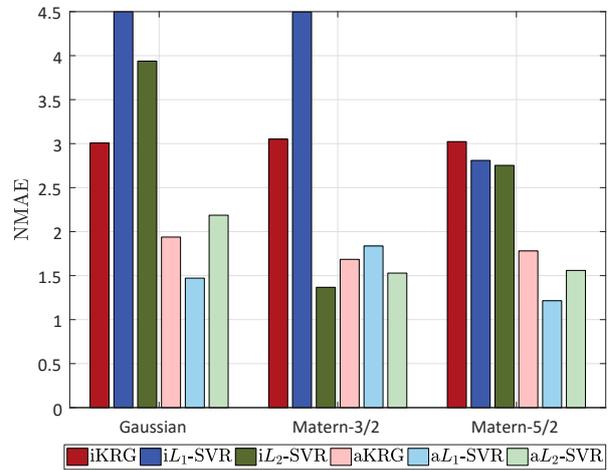
(d) $NMAE$ with $n = 200$
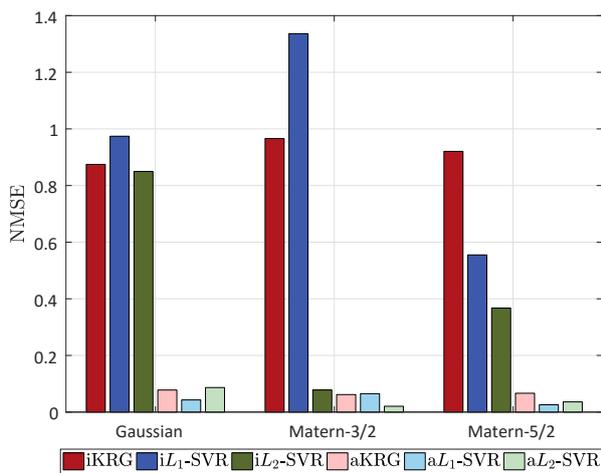
(e) $NMSE$ with $n = 250$
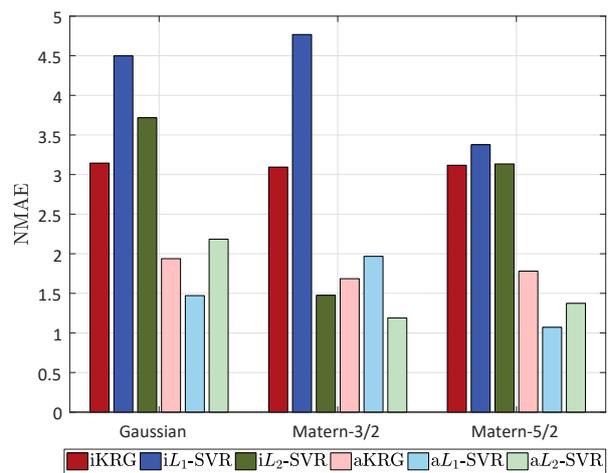
(f) $NMAE$ with $n = 250$

Figure 8: Sobol function - Median values of the normalized mean square and normalized maximum absolute errors.
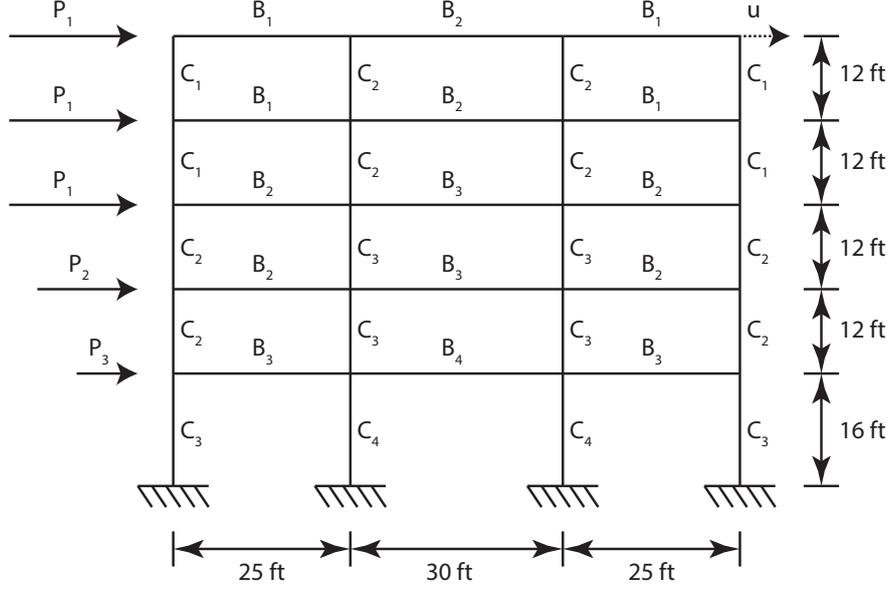
Figure 9: 3-span, 5-story structural frame (Blatman and Sudret, 2010a).

different results shows that the best models with respect to $NMSE$ and $NMAE$ are found with anisotropic Matérn L_1-SVR and anisotropic Gaussian L_2-SVR, respectively. This is illustrated in Fig. 11 where the predictions are plotted against the true values on a sample set of size 1000.

This example proves to be relatively easy to approximate. To have a sharper distinction between the metamodels, we run again the comparison by reducing the experimental size to 100. The results are shown in Fig. 12. Even with this extremely scarce experimental design, results are accurate enough. However, the difference between isotropic and anisotropic models is more important, especially between the SVR models. The accuracy improvement brought by anisotropy in the case of Kriging is actually not too high. The best model with respect to $NMSE$ and $NMAE$ is obtained with anisotropic Gaussian L_2-SVR. Some samples are plotted in Fig. 13a. In contrast, the worst model with respect to $NMSE$ and $NMAE$ is obtained using isotropic Kriging with Matérn 3/2. The corresponding predictions are plotted in Fig. 13b against the true responses.

5.2 The sidemember subsystem

5.2.1 Context of the study

The second example deals with crashworthiness analysis in a context of lightweight design of car bodies. More specifically, the *body-in-white* of an automotive, which is made of metal sheets welded together, has great potential for weight savings. Indeed, when designing a car, engineers aim at finding an optimal distribution of the thicknesses of the body structure

(a) $NMSE$	(b) $NMAE$
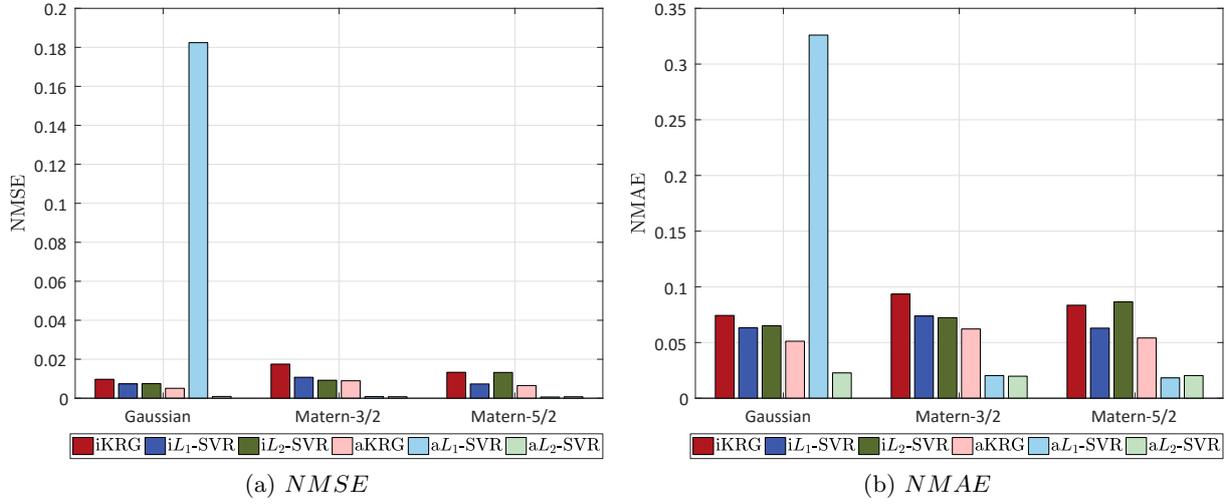
Figure 10: Frame: Lateral displacement - Normalized mean square and normalized maximum absolute errors with $n = 214$.

|	
(a) Best with respect to $NMSE$	(b) Best with respect to $NMAE$
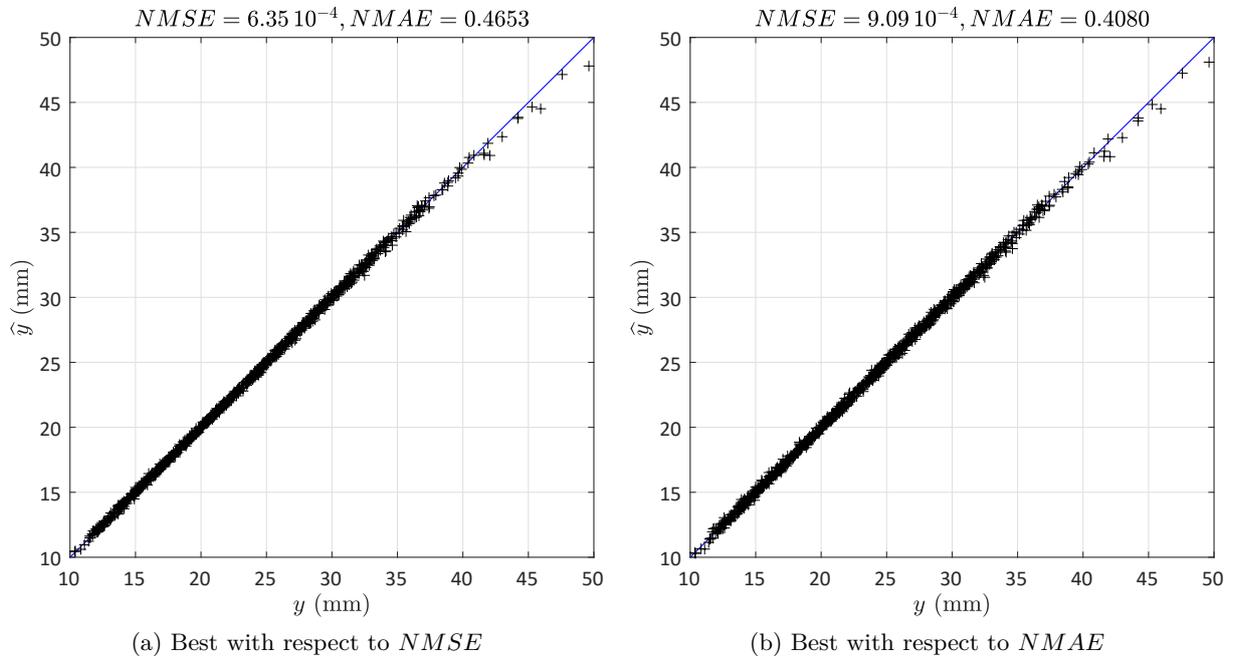
Figure 11: Frame: Best results with respect to $NMSE$ and $NMAE$. Predicted against actual model responses for $n = 214$.

(a) $NMSE$ (b) $NMAE$
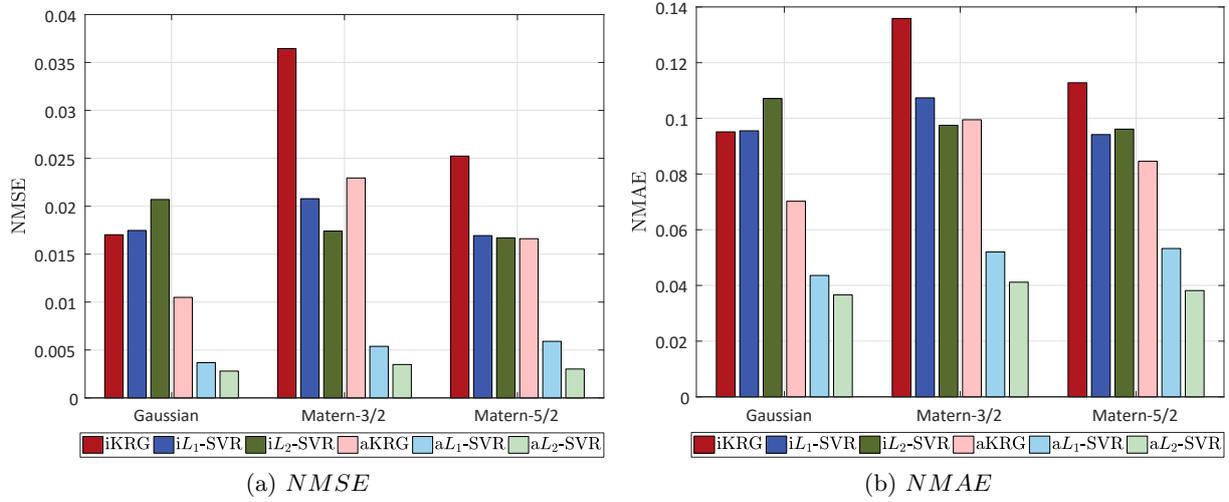
Figure 12: Frame: Lateral displacement - Normalized mean square and normalized maximum absolute errors with $n = 100$.

(a) Best model: Anisotropic L_2-SVR, Gaussian   (b) Worst model: Isotropic Kriging, Matérn 3/2
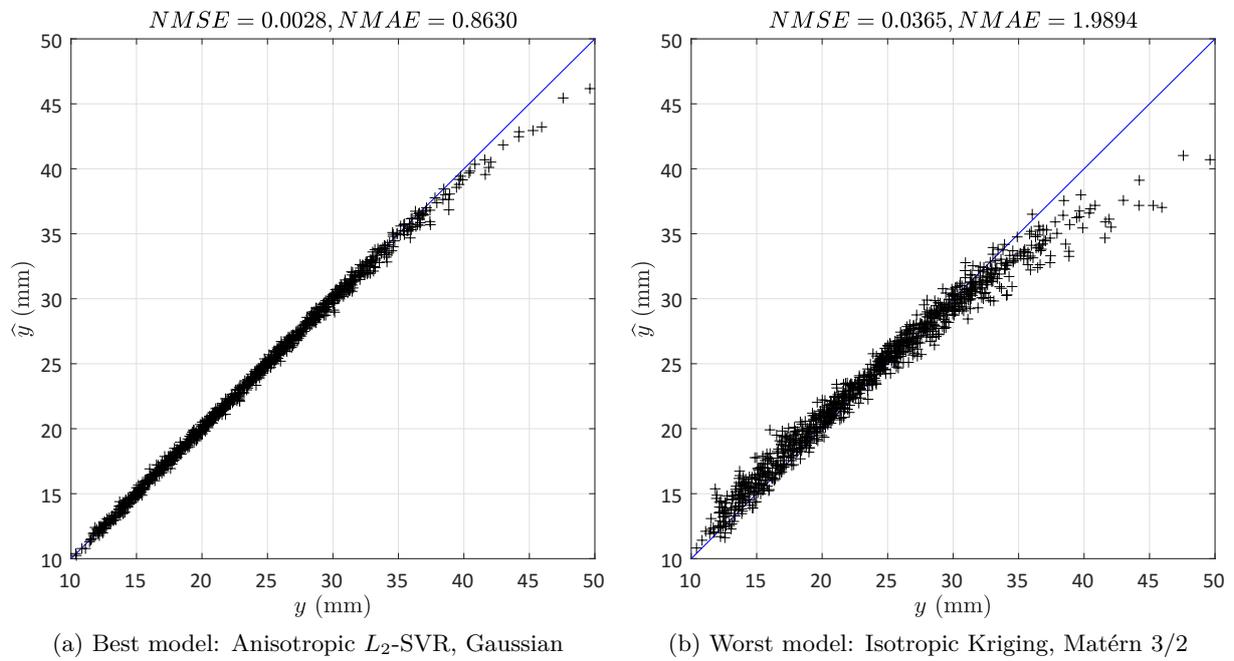
Figure 13: Frame: Best and worst results with respect to $NMSE$. Predicted against actual model responses for $n = 100$.

with minimal weight while respecting some requirements. These are related to security and comfort of passengers *e.g.* acceptable level of noise and vibration in the passenger compartment or sufficient protection of the occupants in an event of crash.

These requirements are usually translated into constraints simulated by the finite element (FE) method, *e.g.* Noise Vibration and Harshness (NVH) analyses and frontal or side impact. Unfortunately, these FE simulations are extremely time-consuming (up to 20 hours on clustered PCs for each single analysis). This, in conjunction with an iterative scheme of optimization, is incompatible with the project timeline, making the methodology unaffordable. Thus one has to resort to surrogate models. However some simulations such as the frontal impact are hardly replaceable by metamodels. The sought outputs are actually highly non-linear and are subjected to noise as they result from a chaotic phenomenon. In other words, bifurcations in the crash simulation can be triggered and exacerbated by small perturbations in the initial conditions, leading to noticeable different crash scenarios (Thole and Mei, 2003; Roux et al., 2006). Moreover, the number of metal sheets involved in an optimization is usually large, making the problem high-dimensional and more difficult to solve.

These are the main three problems encountered when building metamodels for crashworthiness design. Kriging and SVR may be good candidate surrogate models to cope with the above-mentioned features. They are capable of catching non-linearities in the input/output relationship. Besides, the former features the nugget effect for noise handling and a measure of its epistemic uncertainty which is useful in an adaptive experimental design based optimization process (*e.g. Efficient global optimization* (Jones et al., 1998) and related methods). The latter features the ε-insensitive parameter for noise handling.

5.2.2 Presentation of the sidemember subsystem

As an application of the study presented above, a so-called *sidemember subsystem* is introduced. The sidemember is a critical part in the engine compartment structure. It is meant to absorb a high amount of kinetic energy during a frontal impact by properly crushing along its longitudinal axis. It is usually made as an extruded hollow section with strategically positioned mechanical fuses. The sidemember subsystem introduced in this article is a set of parts consisting of the sidemember itself, some parts around it and other impact absorbing parts such as the bumper. An illustration is given in Fig. 14a. Fig. 14b represents an entire car after crash: one can see regions of the front end from which the sidemember subsystem is extracted. Thus the latter leads to a finite element model of smaller size which is far less time-consuming. Nonetheless, it presents approximately the same characteristics as a full car frontal impact *i.e.* high non-linearity and noise.

(a) Sidemember subsystem   (b) Entire model crash simulation

Figure 14: Sidemember subsystem.

5.2.3 Surrogates models and results analysis

The thicknesses of five parts are considered as inputs of the crash simulation, as shown in Fig. 14a. These are the forward sidemember (x_1), the rear sidemember (x_2), the lower bulkhead (x_3), the forward sidemember base (x_4) and the wheel arch (x_5). The range of their thicknesses is $[0.5, 2.95]$, standing for the minimum and maximum available sheet metals thicknesses (in mm). The learning set for this example consists of 150 points of a 5-dimensional Sobol' design. The testing set consists of an already available database consisting of a 500-point Halton sequence. A large set of output quantities can be obtained through simulation. We select a few of them according to the type of output (displacement, force or speed) and the intensity of the noise observed in those outputs. These are respectively the sidemember compression, the wall contact force and the left and right sidemember forces.

The results for these outputs are shown in Figures 15 to 18. According to the selected output, anisotropy may increase significantly (maximum right and left sidemember forces, respectively in Figures 17 and 18) or not (maximum sidemember compression and wall contact force respectively in Figures 15 and 16) the metamodels accuracy. In general, SVR is slightly better than Kriging. Once again, there are some cases when the optimization of the SVR hyperparameters completely fails, leading to spurious results. This mainly happens when using the Gaussian kernel with isotropic L_1-SVR or anisotropic L_2-SVR.

For all the outputs, the best results were globally obtained using anisotropic L_2-SVR with Matérn kernels. Fig. 19 illustrates those best results by plotting the predictions against the actual responses as computed on the validation set. The resulting values of $NMSE$ and $NMAE$ are also indicated on the figures. They respectively range from 0.0028 to 0.0913 and from 0.3017 to 2.3666. While the right sidemember subsystem is the most easily approximated, the left sidemember force exhibits large errors. For the latter, even though the $NMSE$ is not so large compared to the other models, its $NMAE$ is quite important.

28

(a) *NMSE*   (b) *NMAE*
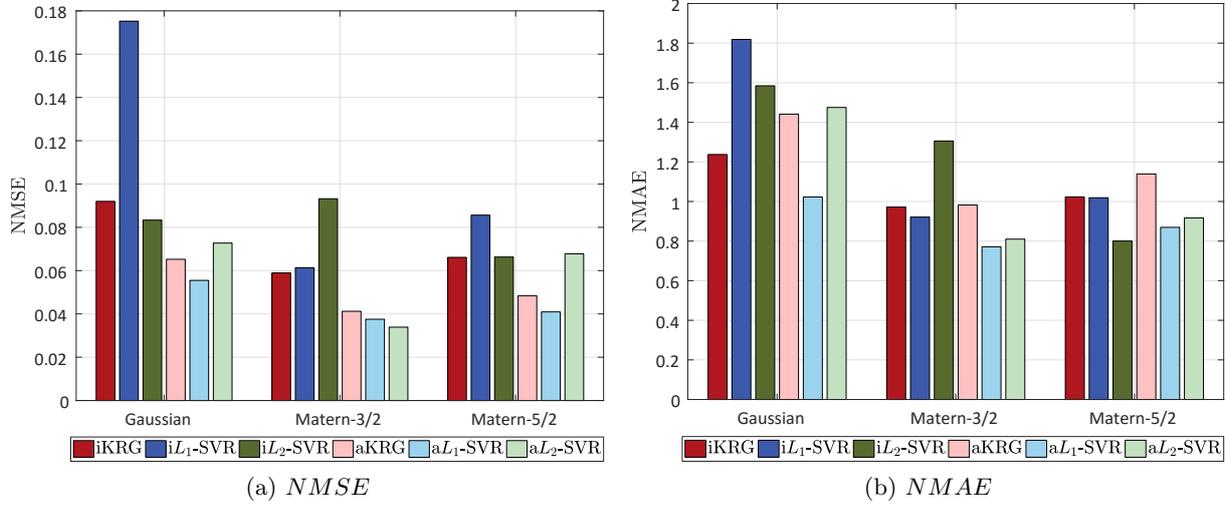
Figure 15: Subsystem - Maximum sidemember compression - Normalized mean square and normalized maximum absolute errors.
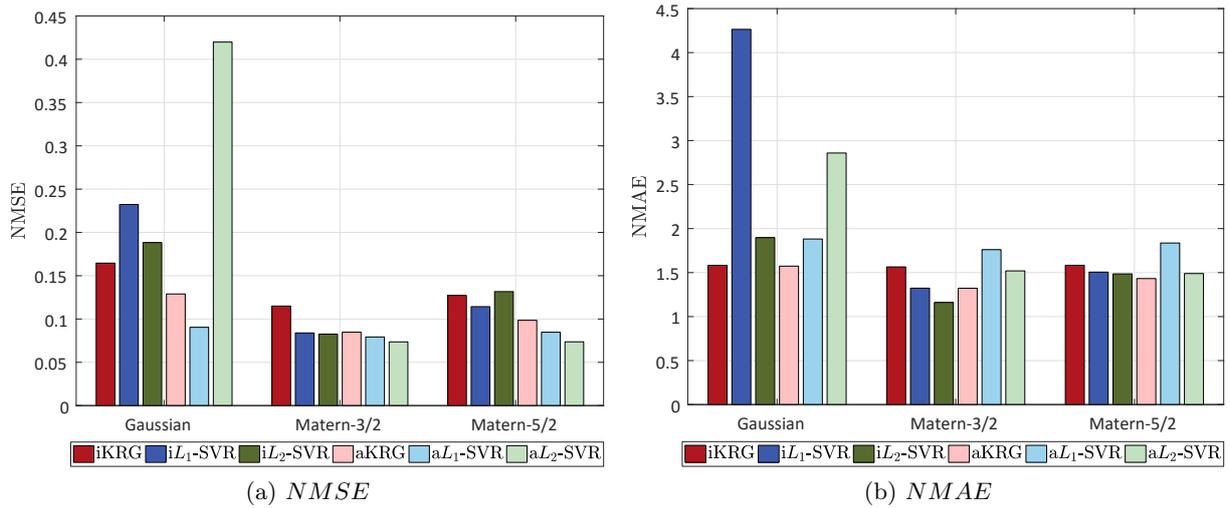
(a) *NMSE*   (b) *NMAE*

Figure 16: Subsystem: Maximum wall contact force - Normalized mean square and normalized maximum absolute errors.
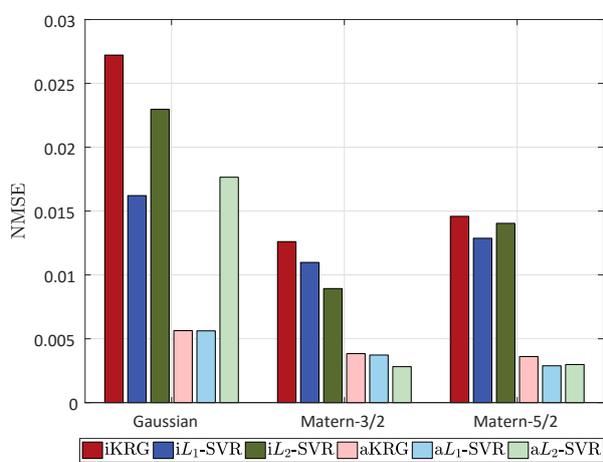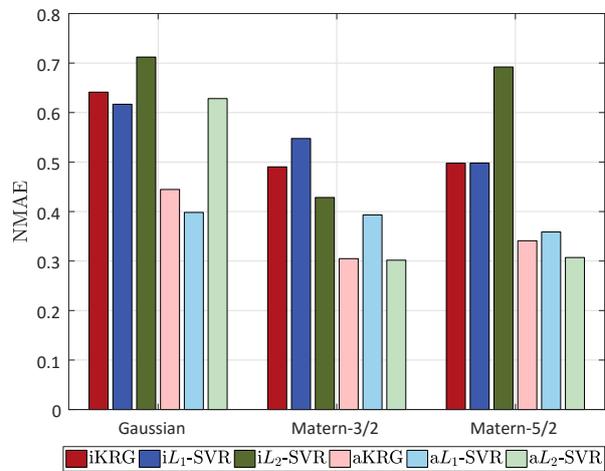
|	
(a) *NMSE*	(b) *NMAE*

Figure 17: Subsystem: Maximum right sidemember force - Normalized mean square and normalized maximum absolute errors.
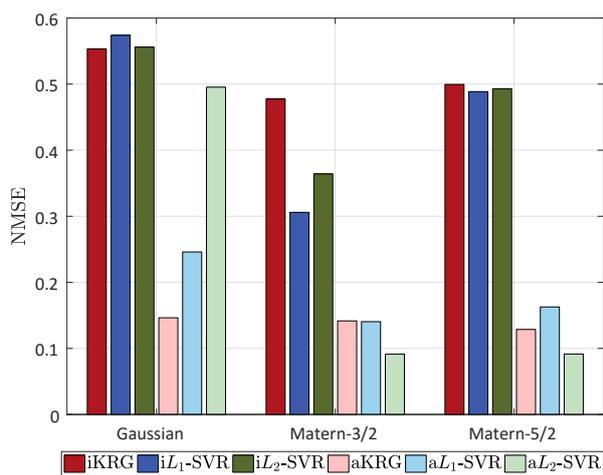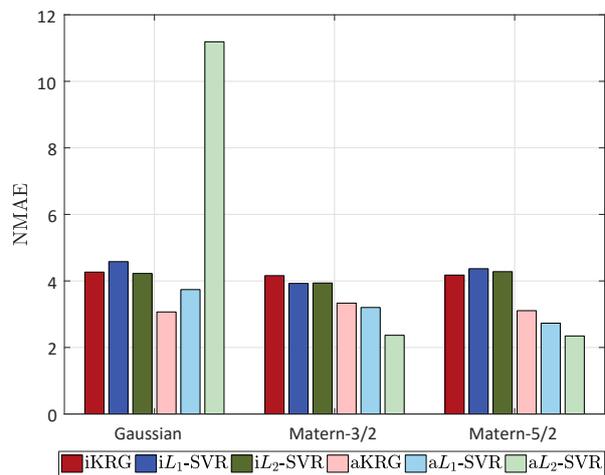
|	
(a) *NMSE*	(b) *NMAE*

Figure 18: Subsystem: Maximum left sidemember force - Normalized mean square and normalized maximum absolute errors.

This is because the extreme upper values are not well approximated as explained by the few points which depart completely from the upper diagonal in Fig. 19d. This is indeed the type of local error (or outliers) that NMAE allows one to identify.

6   Conclusion

In this article, it is proposed to investigate the application of Kriging and support vector regression to the emulation of non-linear, possibly noisy functions. The theoretical background for these two classes of metamodels is first introduced. A unified framework is presented for the calibration of their respective hyperparameters. Both Kriging and SVR are considered with isotropic or anisotropic kernels. Finally, CMA-ES is introduced as a mean to efficiently solve the resulting optimization problems. A comparative study of different forms of these metamodels is then carried out. This involves L_1-SVR, L_2-SVR and Kriging with three different kernels or auto-correlations functions (Gaussian, Matérn 3/2 and 5/2). Three global and local error metrics are considered to assess the accuracy of each metamodel. Some global trend could be identified. First, the introduction of anisotropy dramatically increases the ability of the metamodel to approximate the functions. This is expected as additional degrees of freedom are given to the metamodels. Even though the dimension of the problem is increased, using a robust optimization algorithm allows us to effectively find some optimal hyperparameters, thus leading to extremely accurate models. Despite the use of such an algorithm, there were some failed cases, mostly when using anisotropic SVR with the Gaussian kernel. This may be explained by ill-conditioned matrices that do not allow one to properly estimate the leave-one-out error. Overall, introducing anisotropy and thoroughly calibrating the hyperparameters increases SVR accuracy. In fact, the performance of any metamodeling approach is intrinsically linked to a proper calibration of its hyperparameters, a feature which has been often overlooked in previous comparative studies involving SVR. In this paper, the performance of SVR is improved to the standard of Kriging which was consistently shown to be of higher accuracy in previous comparative studies. Finally, most of the examples were considered with increasing sizes of the experimental design. As expected the errors decrease as the experimental design size increases. However, the optimal experimental design size cannot be known a priori as it depends on factors such as the regularity or the shape of the function to approximate. The configuration that is of interest for real applications is the one where the experimental design is scarce, *i.e.* the number of samples is extremely low compared to the problem's dimension. In such cases, anisotropic L_2-SVR with Matérn has shown overall to be consistently the best model.
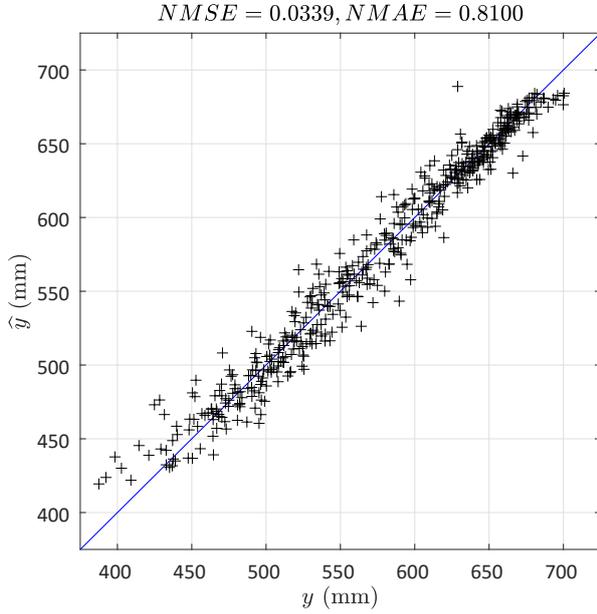
(a) Sidemember compression (anisotropic L_2-SVR, Matérn 3/2)
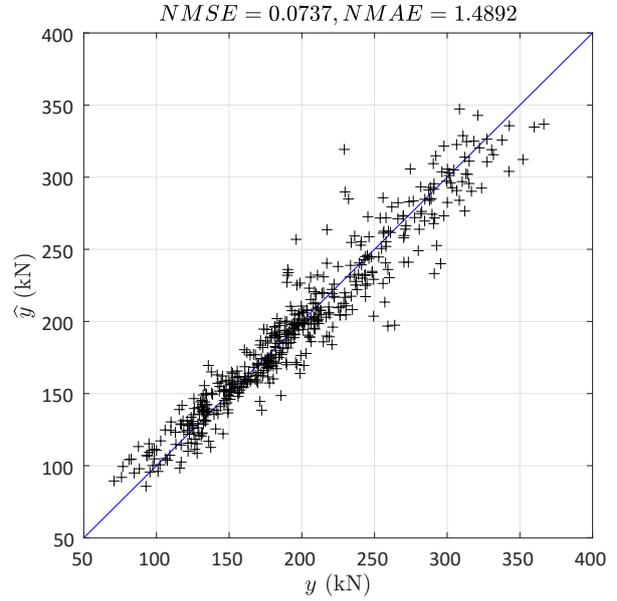
(b) Wall contact force (anisotropic L_2-SVR, Matérn 5/2)
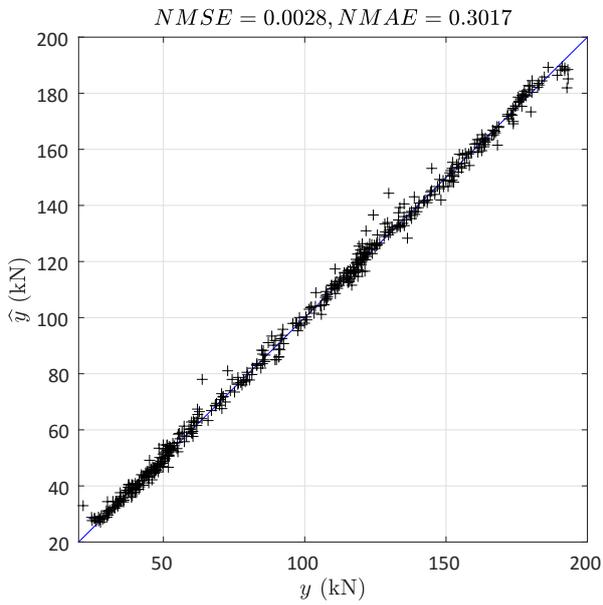
(c) Right sidemember force (anisotropic L_2-SVR, Matérn 3/2)
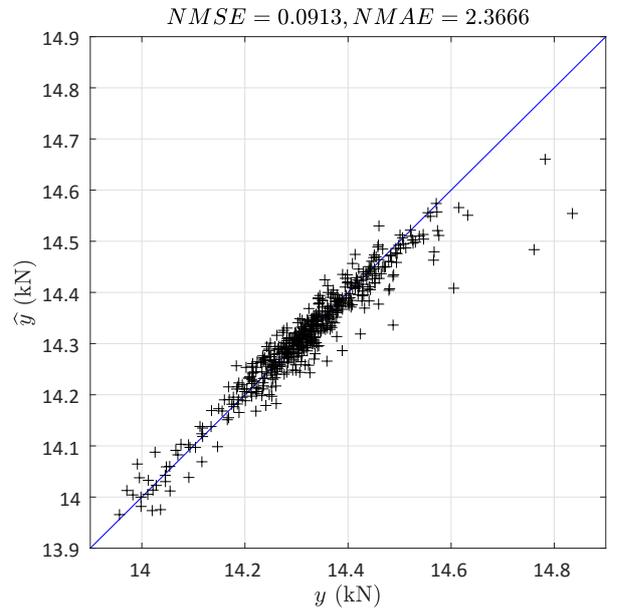
(d) Left sidemember force (anisotropic L_2-SVR, Matérn 3/2)

Figure 19: Subsystem: Best results with respect to $NMSE$. Predicted against actual model responses.

References

Aizerman, M. A., E. A. Braverman, and L. Rozonoer (1964). Theoretical foundations of the potential function method in pattern recognition learning. In *Autom. Rem. Contr.*, pp. 821–837.

Anoop, A. M. and A. Messac (2006). Metamodeling using extended radial basis functions: a comparative approach. *Engineering with Computers 21*, 203–217.

Asmussen, S. and P. W. Glynn (2007). *Stochastic simulation: algorithms and analysis.* Springer.

Au, S.-K. and J. L. Beck (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Prob. Eng. Mech. 16*, 263–277.

Bachoc, F. (2013). Cross validation and maximum likelihood estimations of hyper-parameters of Gaussian processes with model misspecifications. *Comput. Stat. Data Anal. 66*, 55–69.

Balesdent, M., J. Morio, and J. Marzat (2013). Kriging-based adaptive importance sampling algorithms for rare event estimation. *Structural Safety 44*, 1–10.

Blatman, G. and B. Sudret (2008). Adaptive sparse polynomial chaos expansions - application to structural reliability. In *Proc. 4th Int. ASRANet Colloquium, Athens, Greece.*

Blatman, G. and B. Sudret (2010a). An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. *Prob. Eng. Mech. 25*, 183–197.

Blatman, G. and B. Sudret (2010b). Reliability analysis of a pressurized water reactor vessel using sparse polynomial chaos expansions. In D. Straub, L. Esteva, and M. Faber (Eds.), *Proc. 15th IFIP WG7.5 Conference on Reliability and Optimization of Structural Systems, Munich, Germany*, pp. 9–16. Taylor & Francis.

Bompard, M. (2011). *Modèles de substitution pour l'optimisation globale de forme en aéro-dynamique et méthode locale sans paramétrisation.* Ph. D. thesis, Université de Nice Sophia-Antipolis, France.

Bourinet, J.-M. (2015). Reliability assessment with adaptive surrogates based on support vec tor machine regression. In G. S. e. M. Papadrakakis, V. Papadopoulos (Ed.), *Proc. 1st ECCOMAS Thematic Conference on Uncertainty Quantification in Computational Sciences and Engineering, Crete Island, Greece, May 25-27.*

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

Box, G. E. P. and N. R. Draper (1986). *Empirical model building and response surface.* John Wiley & Sons, Inc.

Broomhead, D. and D. Lowe (1988). Multivariable functional interpolation and adaptive networks. *AIP Conf. Proc. 2*, 321–355.

Chang, M.-W. and C.-J. Lin (2005). Leave-one-out bounds for support vector regression model selection. *Neural Comput. 17*(5), 1188–1222.

Chapelle, O. (2002). *Support Vector Machines : principes d'induction, réglage automatique et connaissances a priori.* Ph. D. thesis, Université Pierre et Marie Curie, France.

Chapelle, O., V. Vapnik, O. Bousquet, and S. Mukherje (2002). Choosing multiple parameters for support vector machines. In N. Cristianini (Ed.), *Mach. Learn.*, Volume 46, pp. 131 – 159.

Cherkassky, V. and F. Mulier (2007). *Learning from data: concepts, theory, and methods.* Wiley.

Cortes, C. and V. Vapnik (1995). Support vector networks. In *Mach. Learn.*, pp. 273–297.

De Rocquigny, E. (2012). *Modelling under risk and uncertainty – An introduction to statistical, phenomenological and computational methods.* Wiley series in probability and statistics. John Wiley & Sons, Chichester.

Deheeger, F. and M. Lemaire (2007). Support vector machine for efficient subset simulations: [2]SMART method. In *Proc. 10th Int. Conf. on Applications of Stat. and Prob. in Civil Engineering (ICASP10), Tokyo, Japan.*

Du, Q., V. Faber, and M. Gunzburger (1999). Centroidal Voronoi tessellations: applications and algorithms. *SIAM Rev. 41*(4), 637–676.

Dubourg, V. (2011). *Adaptive surrogate models for reliability analysis and reliability-based design optimization.* Ph. D. thesis, Université Blaise Pascal, Clermont-Ferrand, France.

Dubourg, V., B. Sudret, and J.-M. Bourinet (2011). Reliability-based design optimization using Kriging and subset simulation. *Struct. Multidisc. Optim. 44*(5), 673–690.

Fang, H., M. Rais-Rohani, Z. Liu, and M. F. Horstemeyer (2005). A comparative study of metamodeling method for multiobjective crashworthiness optimization. *Comput. Struct. 83*, 2121–2136.

Franco, J. (2008). *Planification d'expériences numériques en phase exploratoire pour la simulation des phénomènes complexes.* Ph. D. thesis, Ecole Nationale Supérieure des Mines, Saint-Etienne, France.

Ghanem, R. G. and P. D. Spanos (2003). *Stochastic finite elements: a spectral approach.* Courier Dover Publications.

Ginsbourger, D. (2009). *Multiples métamodèles pour l'approximation et l'optimisation de fonctions numériques multivariables.* Ph. D. thesis, École des Mines de Saint-Étienne, France.

Gunn, S. (1998). Support vector machines for classification and regression. Technical Report ISIS-1-98, Dpt. of Electronics and Computer Science, University of Southampton.

Hansen, N. (2005). The CMA Evolution Strategy: A tutorial.

Hansen, N. and S. Kern (2004). Evaluating the CMA-evolution strategy on multimodal test functions. In *Parallel Problem Solving from Nature - PPSN VIII*, Volume 3242 of *Lecture Notes in Computer Science*, pp. 282–291. Springer Berlin Heidelberg.

Hansen, N. and A. Ostermeier (2001). Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation 9*(2), 159–195.

Hurtado, J. E. (2007). Filtered importance sampling with support vector margin: A powerful method for structural reliability analysis. *Structural Safety 29*(1), 2–15.

Jin, R., W. Chen, and T. W. Simpson (2000). Comparative studies of metamodeling techniques under multiple modeling criteria. *Struct. Multidiscip. O. 23*, 1–13.

Jones, D. R. (2001). A taxonomy of global optimization methods based on response surfaces. *J. Global Optim. 21*(4), 345–383.

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

Kleijnen, J. P. C. (2007). *Design and Analysis of Simulation Experiments* (1st ed.). Springer Publishing Company, Incorporated.

Koehler, J. R. and A. Owen (1996). Computer experiments. In S. Ghosh and C. Rao (Eds.), *Handbook of Statistics*, Volume 13, pp. 261–308. North Holland.

Krige, D. G. (1951). A statistical approach to some basic mine valuation problems on the Witwatersrand. *J. Chem., Metal. and Mining Soc. of South Africa 52 (6)*, 119–139.

Li, Y. F., S. H. Ng, M. Xie, and T. N. Goh (2010). A systematic comparison of meta-modeling techniques for simulation optimization in decision support systems. *Appl. Soft Comput. 10*(4), 1257–1273.

Liu, P.-L. and A. Der Kiureghian (1991). Optimization algorithms for structural reliability. *Structural Safety 9*, 161–177.

Lophaven, S. N., H. B. Nielsen, and J. Sondergaard (2002a). Aspects of the Matlab toolbox DACE.

Lophaven, S. N., H. B. Nielsen, and J. Sondergaard (2002b). DACE – A Matlab Kriging toolbox – Version 2.0. Technical University Denmark, Kongens Lyngby, Denmark.

Marelli, S. and B. Sudret (2014). UQLab: A framework for uncertainty quantification in Matlab. In *Vulnerability, Uncertainty and Risk, Proc. 2nd Int. Conf. on Vulnerability, Risk and analysis management (ICVRAM2014), Liverpool, United Kingdom*, pp. 2554–2563.

Marrel, A., B. Iooss, F. van Dorpe, and E. Volkova (2008). An efficient methodology for modeling complex computer codes with Gaussian processes. *Comput. Stat. Data Anal. 52*, 4731–4744.

Matheron, G. (1971). La théorie des variables régionalisées et ses applications.

McCulloch, W. S. and W. Pitts (1988). Neurocomputing: foundations of research. Chapter A logical calculus of the ideas immanent in nervous activity, pp. 15–27. MIT Press.

McKay, M. D., R. J. Beckman, and W. J. Conover (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics 42*(1), 55–61.

Nik, M. A., K. Fayazbakhsh, D. , Pasini, and L. Lessard (2014). A comparative study of metamodelling methods for the design optimization of variable stiffness composites. *Composite Structures 107*, 494–501.

O'Hagan, A. (2006). Bayesian analysis of computer code outputs: A tutorial. *Reliab. Eng. Syst. Safe. 91*(10-11), 1290–1300.

Picheny, V., D. Ginsbourger, O. Roustant, R. T. Haftka, and N. H. Kim (2010). Adaptive designs of experiments for accurate approximation of a target region. *J. Mech. Design 132*(7).

Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian processes for machine learning (adaptive computation and machine learning).* The MIT Press.

Roustant, O., D. Ginsbourger, and Y. Deville (2012). DiceKriging, DiceOptim: two R packages for the analysis of computer experiments by Kriging-based metamodeling and optimization. *J. Stat. Software 51*(1), 1–55.

Roux, W., N. Stander, F. Günther, and H. Müllerschön (2006). Stochastic analysis of highly non-linear structures. *Int. J. Numer. Meth. Eng. 65*, 1221–1241.

Sacks, J., W. Welch, T. Mitchell, and H. Wynn (1989). Design and analysis of computer experiments. *Stat. Sci. 4*(4), 409–423.

Santner, T., B. Williams, and W. Notz (2003). *The design and analysis of computer experiments.* Springer, New York.

Sasena, M. J. (2002). *Flexibility and efficiency enhancements for constrained global design optimization with Kriging approximations.* Ph. D. thesis, University of Michigan, USA.

Smola, A. J. and B. Schölkopf (2004). A tutorial on support vector regression. *Stat. Comp. 14*(3), 199–222.

Sobol, I. M. (1967). On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Comp. Math. 16*, 784–802.

Sudret, B. (2007). *Uncertainty propagation and sensitivity analysis in mechanical models – Contributions to structural reliability and stochastic spectral methods.* Université Blaise Pascal, Clermont-Ferrand, France. Habilitation à Diriger des Recherches.

Thole, C.-A. and L. Mei (2003). Reasons for scatter in crash simulations results. In *4th European LS-DYNA Users Conference, Ulm, Germany.*

Vapnik, V. (1995). *The nature of statistical learning theory.* New York, NY, USA: Springer-Verlag.

Vapnik, V. (1998). *Statistical learning theory.* Wiley.

Vapnik, V. and O. Chapelle (2000). Bounds on error expectation for support vector machines. *Neural Comput. 12*(9), 2013–2036.

Vazquez, E. (2005). *Modélisation comportementale de systèmes non-linéaires multivariables par méthodes à noyaux et applications.* Ph. D. thesis, Université Paris XI Orsay, France.

Wei, D. and S. Rahman (2007). Stuctural reliability analysis by univariate decomposition and numerical integration. *Prob. Eng. Mech. 22*, 27–38.

Yeh, C.-Y., C.-W. Huang, and S.-J. Lee (2011). A multiple-kernel support vector regression approach for stock market price forecasting. *Expert Systems with Applications 38*(3), 2177 – 2186.

Zhao, D. and D. Xue (2010). A comparative study of metamodeling methods considering sample quality merits. *Struct. Multidiscip. O. 42*, 923–938.