Metamodeling for crashworthiness design

Comparative study of kriging and support vector regression

Conference Paper

Author(s):
Moustapha, Maliki; Sudret, Bruno; Bourinet, Jean-Marc; Guillaume, Benoît

Publication date:
2014

Permanent link:
https://doi.org/10.3929/ethz-a-010238087

UNCERTAINTIES 2014

Proceedings of the 2nd International Symposium on Uncertainty Quantification and Stochastic Modeling
July 7th to July 11th, 2014, Rouen, France

METAMODELING FOR CRASHWORTHINESS DESIGN: COMPARATIVE STUDY OF KRIGING AND SUPPORT VECTOR REGRESSION

Maliki Moustapha[1,2,4], Bruno Sudret[3], Jean-Marc Bourinet[1,2], Benoît Guillaume[4]

[1] Clermont Université, IFMA, Institut Pascal, BP 10448, F-63171 Aubière, bourinet@ifma.fr
[2] CNRS, UMR 6602, Institut Pascal, F-63171 Aubière
[3] ETH Zürich, Institute of Structural Engineering, Chair of Risk, Safety & Uncertainty Quantification, Wolfgang-Pauli-Strasse 15, CH-8093 Zürich, sudret@ibk.baug.ethz.ch
[4] PSA Peugeot Citroën, Centre technique de Vélizy, F-78140 Vélizy-Villacoublay, maliki.moustapha@mpsa.com, benoit.guillaume@mpsa.com

Abstract. *The use of metamodels as surrogates of time-consuming functions has widely spread within the academia and the industry. In this paper, two metamodels are considered for crashworthiness design of an automotive body structure, namely Kriging and support vector regression. Variants of these two metamodels associated with various kernel or auto-correlation functions are first compared on analytical functions. The conclusions of this benchmark analysis are then considered to select the most appropriate ones for application on the so-called* sidemember subsystem*. This is a subsystem of an automotive front end under frontal impact. The outputs to emulate are highly non-linear and noisy. The SVR and Kriging models are shown to produce roughly the same level of accuracy for prediction when considered with isotropic kernels or auto-correlation functions, with a slight advantage to Kriging. Besides, the anisotropy in the auto-correlation functions clearly improves the Kriging surrogates. Some outputs of the crash simulation were however hard to surrogate.*

Keywords. *Kriging, support vector regression, noisy data, crashworthiness design*

1 INTRODUCTION

The exponential increase of computational power has favored the development of very high-fidelity finite element models for industrial problems. A typical example is crash tests in the automotive industry. These high-fidelity models however come with the drawback that they can be very-time consuming so that only a few runs of the model can be affordable. They are hence unusable as such in computationally intensive methods which include optimization, reliability or sensitivity analyses. In this context, 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. In this paper we perform a benchmark analysis for two of them namely *Kriging* and *support vector regression*. The aim is to apply them to high dimensional noisy problems such as crashworthiness design. The paper is organized as follows. The first section introduces the two types of metamodels. Then a comparative analysis is performed on two analytical functions. Finally, an application is made on the so-called *sidemember subsystem* under frontal impact.

2 A BRIEF INTRODUCTION TO KRIGING AND SUPPORT VECTOR REGRESSION

2.1 Kriging surrogate

Coming from geostatistics, Kriging has been originally developed by D.G. Krige as a spatial interpolation method. It has later been adapted to computer experiments by Sacks et al. (1989). In this context, let us consider a set of observations: $\mathscr{D} = \left\{ \left(\boldsymbol{x}^{(i)}, y_i \right), i \in \{1, \dots, n\}, \boldsymbol{x}^{(i)} \in \mathbb{X} \subset \mathbb{R}^s, y_i \in \mathbb{Y} \subset \mathbb{R} \right\}$, where the *outputs* y_i result from an unknown mapping of the n s-dimensional *inputs* $\boldsymbol{x}^{(i)}$. The idea behind Kriging surrogates is to consider the output $y(\boldsymbol{x})$ as a realization of a stochastic process $Y(\boldsymbol{x})$:

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

where $\boldsymbol{\beta} = \left\{ \beta_j, j = 1, \dots, p \right\}$ is a weight vector and $\boldsymbol{f} = \left\{ f_j, j = 1 \dots, p \right\}$ is a collection of regression functions. The first part of Eq. (1), $\mu(\boldsymbol{x}) = \sum_{j=1}^{p} \beta_j f_j(\boldsymbol{x})$, 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 $\mathsf{Cov}\left[Z(\boldsymbol{x}), Z(\boldsymbol{x'})\right] = \sigma^2 R(\boldsymbol{x}, \boldsymbol{x'})$, where σ^2 is the process variance and $R(\boldsymbol{x}, \boldsymbol{x'})$ is the auto-correlation matrix providing the dependency structure.

Estimates of the parameters $\boldsymbol{\beta}$ and σ^2 are derived by maximizing the likelihood that the noise $\boldsymbol{Z} = \boldsymbol{Y} - \boldsymbol{F}\boldsymbol{\beta}$ is Gaussian and read:

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

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

Eventually, the prediction for a new point is defined by requiring it to respect three conditions, namely linearity with respect to the observed data, no bias and minimal variance. Once the associated problem is solved, one can get the mean and variance formulations of the Universal Kriging estimator:

$$\begin{aligned} \mu_{\widehat{Y_0}} &= \boldsymbol{f}\left(\boldsymbol{x}^{(0)}\right)^T \widehat{\beta} + \boldsymbol{r}_0^T \boldsymbol{R}^{-1}\left(\boldsymbol{y} - \boldsymbol{F}\widehat{\beta}\right) \\ \sigma^2_{\widehat{Y_0}} &= \widehat{\sigma^2}\left(1 - \boldsymbol{r}_0^T \boldsymbol{R}^{-1} \boldsymbol{r}_0 + \boldsymbol{u}^T (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F})^{-1} \boldsymbol{u}\right) \end{aligned} \tag{3}$$

where $\boldsymbol{u} = \boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{r_0} - \boldsymbol{f}\left(\boldsymbol{x}^{(0)}\right)$ and $\boldsymbol{r_0}$ is a vector gathering the auto-correlation function computed between $\boldsymbol{x}^{(0)}$ and each point of the set of observations $\mathscr{D}$.

This formulation has many features. First, it is interpolating as shown in Vazquez (2005), meaning the prediction is exact at the training points and the associated variance is zero. 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 by suddenly modifying it. This leads to the following modified expression of the auto-covariance of $Y(\boldsymbol{x})$:

$$\boldsymbol{C} = \sigma^2 \boldsymbol{R} + \sigma_\zeta^2 \boldsymbol{I}_n \tag{4}$$

where $\boldsymbol{C}$ is the auto-covariance matrix defined by $\left\{C_{ij} = \mathsf{Cov}\left[Z(\boldsymbol{x}^{(i)}), Z(\boldsymbol{x}^{(j)})\right], i = 1, \dots, n, j = 1, \dots, n\right\}$ and $\boldsymbol{I}_n$ is the identity matrix of size $n \times n$.

Second it is asymptotically consistent (provided that the auto-correlation function is regular) *i.e.* increasing the size of $\mathscr{D}$ decreases the overall variance of the process. Last, the prediction at a given point, say $\boldsymbol{x}^{(0)}$, is considered as a realization of a Gaussian process ($Y_0 \sim \mathscr{N}(\mu_{\widehat{Y_0}}, \sigma^2_{\widehat{Y_0}})$). It is thus possible to derive confidence bounds on a prediction.

2.1.1 Model training

An aspect of prime importance when building a Kriging surrogate is the choice of the auto-correlation function and its parameters. This function relates the assumption about the nature and shape of the system being approximated. Rasmussen and Williams (2005) review some commonly-used auto-correlation functions for Gaussian process modeling. Some simplifying assumptions are usually introduced such as stationarity or isotropy.

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* or *Bayesian estimation.* In computer experiments, the most widely used technique is *maximum likelihood estimation* (MLE) (Koehler and Owen, 1996; Roustant et al., 2012). 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:

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^d} \psi(\boldsymbol{\theta}) = \widehat{\sigma}^2(\boldsymbol{\theta}) \det \boldsymbol{R}(\boldsymbol{\theta})^{\frac{1}{n}} \tag{5}$$

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

This reduced likelihood function is not easy to minimize for various reasons. Lophaven et al. (2002) show how badly conditioned is the auto-correlation function for some values of $\boldsymbol{\theta}$, leading to inaccuracies propagation in the optimization process. $\psi(\boldsymbol{\theta})$ may also have many local minima in the space of the parameters. In the R package *DiceKriging* we are using, Roustant et al. (2012) use a genetic or the second-order BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm.

2.2 Support Vector Regression

Support vector regression is a learning machine technique based on the principle of *structural risk minimization* where, in addition to the training error minimization, the complexity of the metamodel is controlled to avoid overfitting (Vapnik, 1995). This is achieved by mapping the input points into a higher dimensional *feature space* and building the model with the so-called *ε-insensitive loss function* (Cortes and Vapnik, 1995). In this feature space, a linear regressor is sought and deviations of points from this regressor are accepted as long as they are lower than a given threshold, say ε. The coefficients of the model result from the solution of a quadratic convex optimization problem. The following expansion gives the estimation for a new input point $\boldsymbol{x}^{(0)}$:

$$\widehat{y}\left(\boldsymbol{x}^{(0)}\right) = -\sum_{i=1}^{n} (\alpha_i - \alpha_i^*)\, k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(0)}) + b \tag{6}$$

where α_i and α_i^* are Lagrange multipliers from the constrained optimization problem and $k(\bullet,\bullet)$ is the so-called *kernel function* which allows one to carry operations in the feature space without explicitly mapping into it. Examples of well-known kernel functions are polynomials or radial basis functions.

Many comments can be made on this expansion. First, the solution depends only on points for which $\alpha_i - \alpha_i^* \neq 0$. Such points are called *support vectors* and lie outside the ε-insensitive tube. The points which lie on the boundaries of the tube are called *unbounded support vectors* and can be used to compute the offset parameter b:

$$b = \text{sign}(\alpha_i - \alpha_i^*)\varepsilon + y_i + \sum_{j=1, j\neq i}^{n} \left(\alpha_j - \alpha_j^*\right) k(\boldsymbol{x}^{(j)}, \boldsymbol{x}^{(i)}) \quad \forall i : \left\{0 < \alpha_i < C \quad \text{or} \quad 0 < \alpha_i^{(*)} < C\right\} \tag{7}$$

where C is a penalty parameter used in the penalizing of errors.

This penalizing with ε-insensitive loss can either be linear or quadratic resulting respectively to L_1-SVR or L_2-SVR. Many other loss functions such as Huber, Laplace or quadratic and other formulations of SVR exist. In this article, we limit our study to the ε-insensitive loss function and its two formulations.

2.2.1 Model training

The generalization ability of an SVR model relies on a good choice of its parameters: the penalty term C, the insensitive tube width ε and the kernel parameters. They are usually estimated through cross-validation or leave-one-out (LOO) methods where an error measure is minimized solely on the training points. A classical LOO computation would however be unaffordable when the training sample is large. In the past few years, many researchers have developed bounds on the LOO error that are much cheaper to compute. One of them is the *Span bound* developed by Vapnik and Chapelle (2000) for classification. Chang and Lin (2005) derived this bound for regression. The main asset of the span bound of the LOO error is that it directly results from the model construction. The only additional operation is a matrix inversion. However, this estimate is not a continuous function with respect to the model parameters. Chapelle et al. (2002) smoothed it by introducing a regularization term. 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 this error bound and we find it is not necessary to smooth the estimates for proper optimization. This results in a more robust optimization but also requires more time.

3 COMPARATIVE STUDY

3.1 Description of the study

The L_1 and L_2-SVR models introduced above are implemented within MATLAB. The Interior-Point-Convex of the QPC package (Quadratic Programming in C, from the SPM project) is used to solve the quadratic optimization problem. The Span estimate is computed following the works by Chapelle (2002) and Chang and Lin (2005). Eventually, the optimization problem is solved according to the cross-entropy method (Rubinstein and Davidson, 1999). Alternatively, the isotropic and anisotropic Kriging surrogates are used as proposed in the DiceKriging package where the MLE of the parameters is carried with the BFGS algorithm. The homogeneous nugget estimation is also enabled, on one hand for the regularization of the optimization problem and on the other hand for dealing with noisy data. Three different functions are considered as auto-correlation or kernel functions: Gaussian, Matérn 3/2 and Matérn 5/2. The metamodels are compared with respect to the *normalized mean square error*. It is computed on a testing set and reads:

$$NMSE = \frac{\sum_{i=1}^{n_{test}} \left(f(\boldsymbol{x}^{(i)}) - y_i\right)^2}{\sum_{i=1}^{n_{test}} \left(\bar{\boldsymbol{y}} - y_i\right)^2} \tag{8}$$

where n_{test} is the size of the testing set, y_i and $f(\boldsymbol{x}^{(i)})$ are respectively the actual and estimated outputs at the point $\boldsymbol{x}^{(i)}$ and $\bar{\boldsymbol{y}}$ is the mean value of the actual outputs in the testing set.

Two analytical functions of different dimensions are used to generate the training and testing samples. The training samples are generated on an hypercube following three different designs of experiments: a Sobol' design, a Centroidal Voronoi Tessellations design and an optimized latin hypercube.

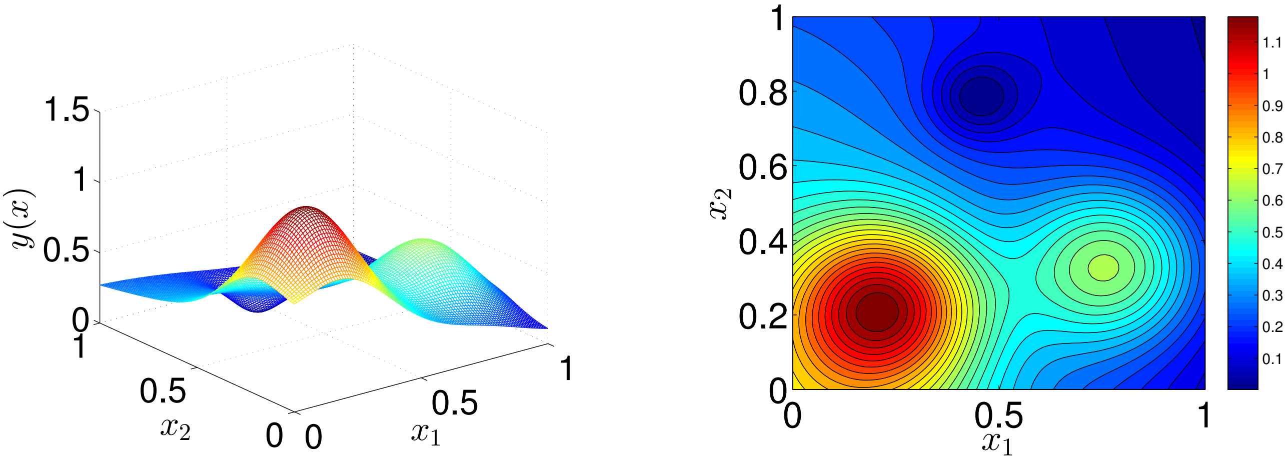
3.1.1 Franke's function

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

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

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

This function is highly non-linear and features two Gaussian peaks and a local minimum as illustrated in Fig. 1 below. For each design, three sizes are considered: 10, 20 and 40. The testing set consists of a Monte-Carlo of 10^6 points.



(a) Three-dimensional representation (b) Level sets

Figure 1: Representation of the two-dimensional Franke's function.

The results for $n = 10$ are shown in Fig. 2 below. On this example, Kriging clearly outperforms SVR. The same levels of $NMSE$ are approximately found when switching the designs, with a slight disadvantage for the optimized latin hypercube. Anyway, we are not intending to compare the designs on this example since we are in a low dimensional space and any random design with a few points *luckily* positioned would perform very well. Finally, the SVR models with the Gaussian kernel produce very large $NMSE$ if compared with other cases. A similar behavior was found with other low-dimensional testing functions with scarce samples, where the optimization failed to find to best hyperparameters.

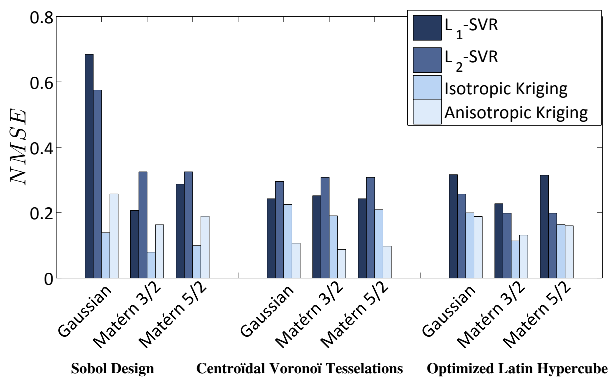


Figure 2: Normalized mean square error for 10 points designs on the Franke's function.

The benchmark with 20 points shows approximately the same trend as for 10 points. Kriging still gives better results than SVR. However, the difference is not as large. The Gaussian kernel SVR with Sobol' design is wrong again.

For 40 points, we have very low levels of $NMSE$ for all configurations. Results are gathered in Tab. 1. For a given kernel, there is no significant difference between L_1-SVR and L_2-SVR on one hand and anisotropic and isotropic Kriging on the other. Globally, Kriging is better than SVR with the Gaussian auto-correlation whereas SVR slightly outperforms Kriging with both Matérn kernels.

Table 1: Normalized mean square error ($NMSE$) for the 40 points designs.

Design	Sobol'			CVT			OLH		
Model	Gauss	Mat3/2	Mat5/2	Gauss	Mat3/2	Mat5/2	Gauss	Mat3/2	Mat5/2
L_1-SVR	0.1894	0.0051	0.0034	0.0094	0.0040	0.0083	0.0396	0.0024	0.0027
L_2-SVR	0.0049	0.0051	0.0034	0.0042	0.0040	0.0018	0.0036	0.0024	0.0025
Iso Kriging	0.0030	0.0122	0.0061	0.0034	0.0049	0.0027	0.0033	0.0057	0.0041
Aniso Kriging	0.0029	0.0128	0.0062	0.0034	0.0049	0.0029	0.0033	0.0058	0.0041

3.1.2 *Sobol' g-function*

This second example is concerned with the Sobol' g-function. It is a function widely used in sensitivity analyses as its sensitivity indices can be expressed analytically. We however use it here for model surrogating since it is fairly complex and its parameters can be tuned to control the importance of a given dimension. It reads:

$$y(\boldsymbol{x}) = \prod_{i=1}^{s} \frac{|4x_i - 2| + a_i}{1 + a_i} \tag{10}$$

where $s = 20$ 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 very important when $a_i = 0$ and becomes insignificant for $a_i \geq 100$. In this paper, we choose to have some variables non-important as that is most likely to happen in real physical high-dimensional problems. Besides, we alternate the level of importance between the dimensions so that we have:

$$a_i = \begin{cases} 3(i-1) & \text{if } i \text{ is odd} \\ 30i & \text{if } i \text{ is even} \end{cases} \tag{11}$$

The testing set is reduced to a Monte Carlo of 10^5 points whereas the training sets consist of 150 points each. Figure 3 shows the different results. The $NMSE$ are very high for the isotropic models. The global trend is that SVR is better than isotropic Kriging on the Gaussian and Matérn 5/2 kernels. Kriging with Matérn 3/2 clearly improves the accuracy of the prediction. Finally, the introduction of anisotropy dramatically decreases the level of the computed $NMSE$. Overall, the anisotropic Kriging surrogates are the best.

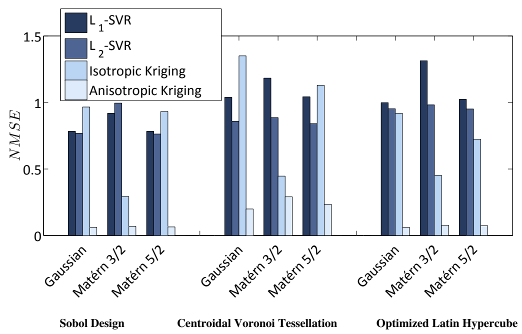


Figure 3: Normalized mean square error for 150 points designs on the Sobol' g-function.

4 APPLICATION TO THE SIDEMEMBER SUBSYSTEM

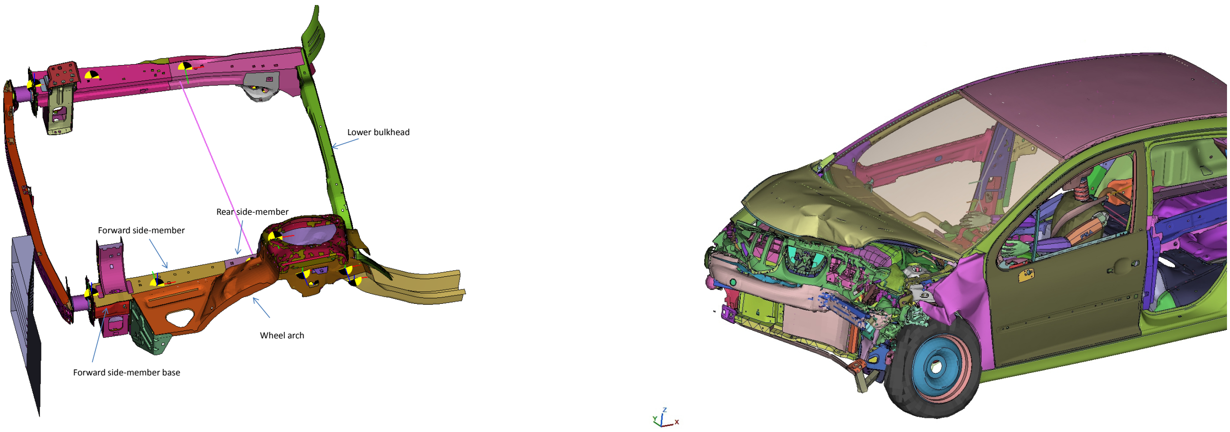
4.1 Context and presentation of the sidemember subsystem

As stated in introduction, we are aiming at building metamodels as surrogates of crashworthiness related functions for automotive body structures lightweight design. In this context, the constraints of the problem are very time-consuming, highly-non linear and subjected to noise as they result from a chaotic phenomenon. That is, bifurcations in the crash simulation can be triggered and exacerbated by small perturbations in the initial conditions, leading to noticeable different crash scenarios.

In this article, Kriging and support vector regression are applied to approach outputs of crash simulations of the so-called *sidemember subsystem*. It is a set of parts consisting of the sidemember itself with some parts around it and other impact absorbing parts such as the bumper. An illustration is given in Fig. 4a. Figure 4b is a representation of an entire car after crash and one can see regions of the front end from which the sidemember subsystem is extracted.

4.2 Surrogates models and results analysis

Five parts are considered as inputs as shown in Fig. 4a above. 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).



(a) Sidemember subsystem (b) Entire FE model of a car after crash

Figure 4: The sidemember subsystem.

The range of their thicknesses is $[0.5, 2.95]$, standing for the minimum and maximum available metal sheets thicknesses (in mm). The learning set for this example consists of 150 points of a 5-dimensional Sobol' design. The testing set consists of the first 500 points of a Halton sequence. A wide set of outputs are available for analysis as post-processing data of the simulation. We selected a few of them according to the type of the output (displacement, force or speed) and associated levels of noise. For each metamodel formulation, one kernel or auto-correlation are selected for this analysis according to the results of the benchmark. The two qualities sought were robustness of the model and accuracy. Considering these two aspects, we kept the Gaussian kernel for L_1-SVR, Matérn 3/2 for L_2-SVR and isotropic Kriging and Matérn 5/2 for anisotropic Kriging. Results for some outputs are presented in Tab. 2 below. First, for all the cases, anisotropic Kriging gives the best results accordingly with the results of the previous section. The results for the SVR and Kriging with Matérn 3/2 are relatively close. Finally, some outputs are more easy to approximate by the metamodels than others. A closer look at the outputs might give a hint on the difficulties encountered while building the surrogates.

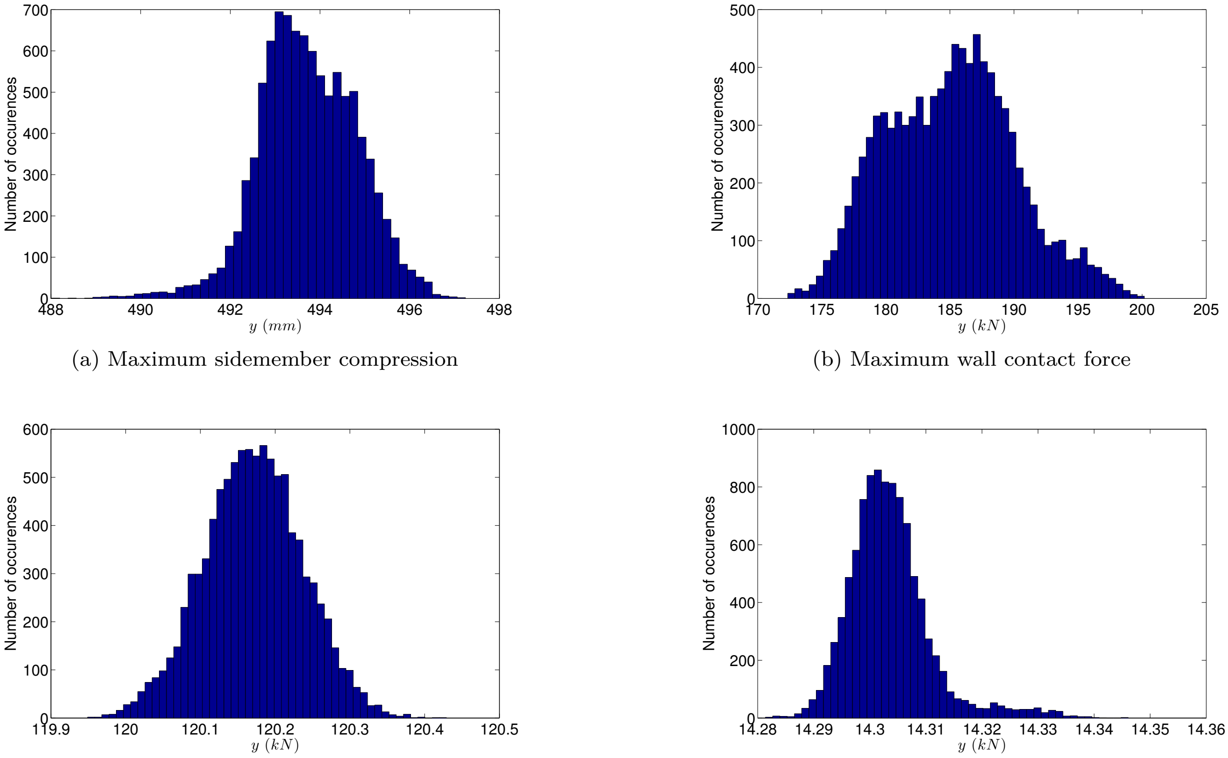
Table 2: Normalized mean square errors for some of the sidemember subsystem outputs

Model	L_1-SVR	L_2-SVR	Iso KRG	Ani KRG
Sidemember compression	0.1381	0.0525	0.0507	0.0359
Max. wall contact force	0.3864	0.0967	0.0813	0.0746
Max. right fwd sidemember force	0.9159	0.4041	0.3816	0.1139
Max. left fwd sidemember force	0.0316	0.0136	0.0089	0.0034
Sidemember speed variation	0.3343	0.1642	0.1403	0.0957

Let us first consider the noise in the data for these outputs. Usually to assess the level of noise in frontal impact simulations, so-called *numerical* and *physical scatter* analyses are carried out. The former consists of a set of simulations with the same initial conditions where the coordinates of the FE mesh are randomly and infinitesimally perturbed. The idea is to assess how the numerical instabilities of the solver amplify the slightest variations of the initial conditions. A scatter analysis consisting of 10000 simulations at the nominal values is hence performed. Figure 5 shows histograms of the outputs for the four first functions in Tab. 2. First observation is that the distribution looks Gaussian. However, a Kolmogorov-Smirnov test in Matlab rejects the hypothesis of a normal distribution. Second, lower scatter on an output do not always mean good accuracy of the metamodels.

Furthermore, let us consider the physical scatter analysis on the sidemember subsystem. This analysis is simply carried by applying a random noise on the input parameters of the nominal design (both design variables such as the metal sheets thicknesses and environment variables such as the initial speed of the car for instance). It aims at evaluating the physical robustness of the model. Figure 6 below shows the evolution of the left and right forward sidemember force corresponding to 100 simulations of the nominal design with a slight perturbation of the initial conditions. The resulting scatter of the outputs is relatively high. For the metamodels, the sought responses are the maximum of each curve. One can see for the right forward sidemember force that the maximum actually occurs at totally different times. This means that the crash scenarios somehow change. This may explain difficulties in properly approximating this output. This however remains an hypothesis and needs to be verified with further analyses.

On a bigger scale, plotting the time of occurrence of a sought response *vs.* its value shows clusters that represent different solution paths. On the five outputs above, two exhibit these clusters: the right forward sidemember force and the maximum wall contact force. Figure 7 illustrates the latter case. One can identify two or three clusters of points with respect to the time of occurrence of the maximum value. This might mean that we are trying to build a single model to surrogate different physical phenomena.



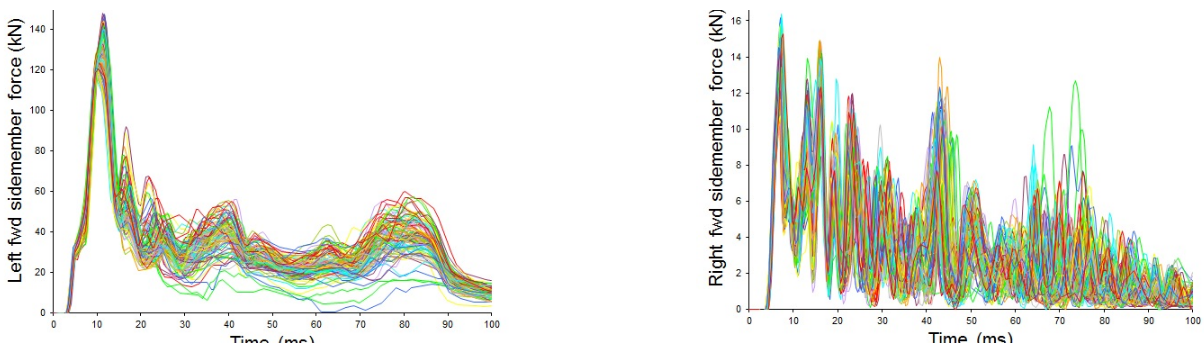
(a) Maximum sidemember compression

(b) Maximum wall contact force

(c) Maximum left forward sidemember force

(d) Maximum right forward sidemember force

Figure 5: Numerical scatter analysis. Histograms of outputs of four functions.



(a) Maximum left forward sidemember force

(b) Maximum right forward sidemember force

Figure 6: Physical robustness analysis. Evolution of the forward sidemember force (left and right).

5 CONCLUSION

In this paper, it was proposed to investigate the application of Kriging and support vector regression to the emulation of non-linear noisy functions. The theoretical background on these two metamodels was first introduced. A comparative study of different forms of these metamodels was then carried out. This involved L_1-SVR, L_2-SVR, anisotropic and isotropic Kriging with three kernels or auto-correlations functions (Gaussian, Matérn 3/2 and 5/2). The investigated cases did not exhibit one systematically best model. However there was some trend. The introduction of anisotropy dramatically increases the ability of the metamodel to approximate the functions. For the isotropic models, L_1-SVR was globally the best model with the Gaussian kernel while Kriging was most of the time better with the Matérn auto-correlation functions. The application to the sidemember subsystem confirmed this trend. The reasons for the poor fitting on this models were analyzed. It will be interesting in a future work to perform the same benchmark on analytical functions with introduction of different levels of noise as in the sidemember subsystem. This will serve as reference for a future application where a metamodel-based robust design optimization of a car body structure under crashworthiness related constraints is performed.

6 ACKNOWLEDGEMENTS

The first author was funded by a CIFRE grant from PSA Peugeot Citroën subsidized by the ANRT.

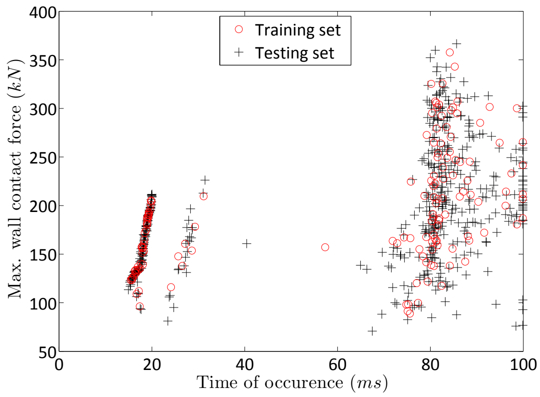


Figure 7: Maximum wall contact force with respect to its time of occurrence during the crash simulation.

REFERENCES

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.

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

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

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

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.

Lophaven, S. N., H. B. Nielsen, and J. Sondergaard (2002). Aspects of the Matlab tollbox DACE. Technical Universty Denmark.

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.

Rasmussen, C. E. and C. K. I. Williams (2005). *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. Softw. 51*(1), 1–55.

Rubinstein, R. and W. Davidson (1999). The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability 1*(2), 127–190.

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

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

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

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

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.