



HAL
open science

Estimation of the derivative-based global sensitivity measures using a Gaussian process metamodel

Matthias de Lozzo, Amandine Marrel

► **To cite this version:**

Matthias de Lozzo, Amandine Marrel. Estimation of the derivative-based global sensitivity measures using a Gaussian process metamodel. 2015. hal-01164215v1

HAL Id: hal-01164215

<https://hal.science/hal-01164215v1>

Preprint submitted on 16 Jun 2015 (v1), last revised 4 Dec 2015 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Estimation of the derivative-based global sensitivity measures using a Gaussian process metamodel

Matthias De Lozzo¹ and Amandine Marrel¹

¹CEA,DEN,DER, F-13108 Saint Paul Lez Durance, France

Abstract

Physical phenomena are often studied using numerical simulators. Such computer codes are function of uncertain input parameters and a global sensitivity analysis (GSA) can be performed to identify their impacts on the simulator outputs. Sobol' indices, based on output variance decomposition, are commonly used to perform quantitative GSA. For many years now, other tools have been studied, closer to physical practices such as the derivative-based global sensitivity measures (DGSM). However, numerical simulators rarely provide the output gradient and DGSM estimation is not directly possible. To address this limitation, we propose to estimate the DGSMs using a Gaussian process metamodel (GPM) which approximates the simulator.

Based on this GPM, we propose two DGSM estimators: a plug-in one defined by the DGSM of the GPM predictor and another one defined by the expectation of the DGSM associated to the full-GPM. The latter is equal to the first one completed by a variance term and can be accompanied by a confidence interval. For Gaussian kernel and uniform input laws, analytical formula are given for both DGSM estimators. For all other situations, Monte-Carlo methods for the expectation approximations are proposed: a propagative version of the Gibbs sampler and a chi-square approximation. Moreover, a significance test for the full-GPM based estimator is proposed for screening. The convergence of the two GPM-based DGSM estimators and the Monte-Carlo approaches are compared on the Ishigami function. Finally, we apply our work to an environmental application, showing the interest of the significance test and discussing the meaning of the DGSMs.

Keywords. Sensitivity analysis, derivative-based global sensitivity measures (DGSM), Gaussian process model, significance test, screening.

AMS subject classification. 62G99, 65C60, 60G15, 62P30.

1 Introduction

Computer models are widely used for the representation of physical phenomena in the industry. The main reasons are the permanent increase of computational power and numerical method efficiency. Such models take as input many numerical and physical explanatory variables. They can be used in computer experimentation to explore the relationship between the variables and increase the knowledge about the physical phenomenon, while an accurate physical experimentation is too costly (Santner et al., 2003). However these numerical simulators often have a high number of uncertain input parameters, what can lead to an important uncertainty over the model output (de Rocquigny et al., 2008; Kleijnen, 1997). Commonly, the input uncertainties are modeled by probability distributions and a quantification of their impacts on the output is performed. These steps define a sensitivity analysis (SA) of the considered computer code. Then, the SA results could be used to reduce the model output uncertainty by enhancing the characterization of the most influential input parameters and using nominal values for the non-significant ones.

This SA can be either local (LSA), when it focus on the output behavior associated to input variations around a specific input parameter value, or global (GSA), when it considers the output behavior for uncertainties based on the whole variation space of the input parameters (Saltelli et al., 2008). GSA methods are widely used for industrial problems involving many variables (Saltelli et al., 2004). They can distinguish the non-significant input parameters and the significant ones in a screening context. They can

also quantify and order the influences of the significant parameters and of their interactions on the model output behavior. Recently, Iooss and Lemaitre (2015) propose a review on GSA in a methodological framework, presenting screening methods, measures of importance and tools for the “deep exploration” of computer code behaviors. Any GSA method has advantages and drawbacks: some approaches require a high number of computer experiments but cover a wide scope of model behaviors, while other ones are less greedy but rely on hypotheses such as a monotonic or linear behavior of the model output with respect to the input variables. The second ones are mainly used for qualitative purposes such as screening, while the first ones are used in a quantitative aim when an influence ranking of the input parameters is wished. In this second case, the most popular are the variance-based indices, usually represented by the Sobol’ index which measures the proportion of the output variance explained by an input parameter (Sobol, 1993), considering either its single contribution (first-order index), or its whole contribution including the interactions with the other input parameters (total index, e.g. Homma and Saltelli (1996)). Recently, Fort et al. (2013) introduced the Goal Oriented Sensitivity Analysis, providing an unified framework for several sensitivity analyses based on the mean output value, on a specific output quantile for excess probability considerations, and so on. With the same idea to go further than the Sobol’ indices, Da Veiga (2014) proposes new sensitivity measures considering the whole distribution of the output respect to those of the input parameters, either comparing characteristic functions (Székely et al., 2007), or measuring the covariance between input and output parameters in some reproducing kernel Hilbert spaces (Gretton et al., 2005). Based on these measures, De Lozzo and Marrel (2015) study their meaning with respect to the Sobol’ indices, and propose extensions to screening. Likewise, Plischke et al. (2013) propose sensitivity indices measuring the distance between the distribution of the model output conditioned by an input parameter with the unconditioned one.

From a more physical point of view, Sobol and Gresham (1995) introduced the derivative-based global sensitivity measure (DGSM) which represents the mean of the squared partial derivative of the model output over the input domain. This approach can be tied with the well-known Morris method (Morris, 1991) which considers finite differences instead of the local derivatives: in a way, DGSMs can be viewed as a generalization of the indices obtained with the Morris method (Touzani and Busby, 2014). The DGSMs often have the advantage of requiring much less model evaluations than the Sobol’ indices (Kucherenko et al., 2009), especially for high-dimensional problems (many tens of input parameters) or complex models (non linear and interaction effects). Moreover, based on output partial derivatives, they provide more local information which can be complementary to the one brought by the Sobol’ indices for example.

Various applications have illustrated the use of DGSMs such as an aquatic prey-predator chain (Iooss et al., 2012), a biological system model (Kiparissides et al., 2009), a flood one (Lamboni et al., 2013) or a reservoir simulator (Touzani and Busby, 2014). For a screening purpose, it has recently been proved that the Sobol’ indices are upper bounded up to a constant by the DGSMs, firstly in the case of uniform or normal probability distributions (Sobol and Kucherenko, 2009), and then in the case of a wider variety of continuous distributions (Lamboni et al., 2013). These sensitivity indices have also been extended to the interaction between two input parameters in (Roustant et al., 2014). The authors call such a measure a crossed DGSM and define it by the mean of the square partial derivative of the output model according to both input variables. They also provide an inequality link between the crossed-DGSMs and the total Sobol’ indices.

For all these reasons, we focus here on the use of DGSMs, especially for a screening purpose. The computation of these sensitivity indices then requires that the model supplies the adjoint code to evaluate its output partial derivatives. When this code is unavailable, the gradient can be estimated by a finite-difference method. However, in presence of few numerical simulator evaluations, this alternative can lead to an important approximation error of the output gradient, which increases the DGSM one associated to the integral quadratures. Another alternative consists in replacing the time-expensive numerical simulator by a surrogate model (Forrester et al., 2008) and using its gradient; this is the choice that we make in this paper. A surrogate model, also called metamodel, is a fast mathematical model built using some evaluations of the numerical simulator in order to approach its behavior. Polynomial chaos expansions (Soize and Ghanem, 2004), artificial neural networks (Dreyfus, 2005) and Gaussian process metamodels (Rasmussen and Williams, 2005) are the most popular ones. Recently, metamodeling techniques have been applied with polynomial chaos expansions to DGSMs (Sudret and Mai, 2014) and Sobol’ indices (Sudret, 2008). Gaussian process metamodels have also been used for the estimation of Sobol’ indices (Marrel et al., 2009); in particular, these surrogate models can provide confidence intervals to quantify the approximation error of such sensitivity indices.

In this paper, following the same approach of Marrel et al. (2009), we investigate the DGSM approximation based on the replacement of the numerical simulator by a Gaussian process metamodel. Moreover, we propose a DGSM-based significance test for a screening purpose, in order to separate the significant inputs from the non-significant ones. We also deal with the meaning of the DGSM from local and global points of view.

Firstly, we introduce the definition of DGSM and Gaussian process metamodel (GPM). Then, we propose two estimators of the DGSMs based on a GPM: one using the GPM predictor and another one considering the whole GPM distribution. In this second section, we also propose a confidence interval for the DGSM, using the probability density function of the metamodel. Then in Section 3, we give explicit formulations of these estimators for a particular covariance function of the GPM and specific input parameter distributions. In all other cases, we propose to estimate the DGSM formulas by several Monte-Carlo methods. Moreover, we build a significance test based on DGSM to reject the non influential inputs for a screening purpose. Finally, the different DGSM estimations and associated computational methods are compared on an analytical model in Section 4. An application to an industrial test is also proposed, for a screening purpose.

2 DGSM formulation using a Gaussian process metamodel

We consider the computer code

$$y = f(x_1, \dots, x_d)$$

where x_1, \dots, x_d are d input uncertain parameters, resumed by the vector $x = (x_1, \dots, x_d)$. The function f maps from $\mathcal{X} = \prod_{k=1}^d \mathcal{X}_k \subset \mathbb{R}^d$ to \mathbb{R} and is assumed to be square integrable. We are interested in the variability of the output $f(x)$ with respect to the different input parameters over their definition domain \mathcal{X} . In the context of Global Sensitivity Analysis (GSA) (Saltelli et al., 2008), the x_1, \dots, x_d are considered as realizations of the independent random variables X_1, \dots, X_d whose probability density functions $\mu_{X_1}, \dots, \mu_{X_d}$ are known, with $\mu_{X_k}(x) > 0$ over \mathcal{X}_k for any $k \in \{1, \dots, d\}$. We want to measure the dependence between the random input parameters and the random output $Y = f(X)$ using sensitivity indices.

In this paper, the GSA is based on a n -sample $(X^{(i)}, Y^{(i)})_{1 \leq i \leq n}$ made of n independent and identically distributed (i.i.d.) computer code runs, where the output value $Y^{(i)} = f(X^{(i)})$ is function of the i^{th} input vector value $X^{(i)} = (X_1^{(i)}, \dots, X_d^{(i)}) \in \mathcal{X}$; $(X^{(i)}, Y^{(i)})$ is called a computer experiment, or a code simulation.

2.1 GSA based on Sobol' indices

A classical approach in GSA consists in computing the first-order and total Sobol' indices which are based on the output variance decomposition (Sobol, 1993; Homma and Saltelli, 1996). If the variables X_1, \dots, X_d are independent and if $\mathbb{E}[f^2(X)] < +\infty$, we can apply the Hoeffding decomposition to the random variable $f(X)$ (Efron and Stein, 1981):

$$\begin{aligned} f(X) &= f_0 + \sum_{i=1}^d f_j(X_j) + \sum_{i=1}^d \sum_{i < j}^d f_{ij}(X_i, X_j) + \dots + f_{1\dots d}(X_1, \dots, X_d) \\ &= \sum_{u \subset \{1, \dots, d\}} f_u(X_u) \end{aligned} \quad (1)$$

where $f_0 = \mathbb{E}[f(X)]$, $f_j(X_j) = \mathbb{E}[f(X)|X_j] - f_0$ and $f_u(X_u) = \mathbb{E}[f(X)|X_u] - \sum_{v \subset u} f_v(X_v)$, with $X_u = (X_i)_{i \in u}$, for all $u \subset \{1, \dots, d\}$. All the 2^d summands in (1) have zero mean and are mutually uncorrelated with each other. This decomposition is unique and leads to the Sobol' indices. These are the elements of the $f(X)$ variance decomposition according to the different groups of input parameter interactions in (1). More precisely for each $u \subset \{1, \dots, d\}$, the first-order and total Sobol sensitivity indices of X_u are defined by

$$S_u = \frac{\mathbb{V}[f_u(X_u)]}{\mathbb{V}[f(X)]} \text{ and } S_u^T = \sum_{v \supset u} S_v,$$

Clearly $100S_u$ is the percentage of the output variance explained by X_u , independently from the other inputs, and $100S_u^T$ is the percentage of the output variance explained by X_u considered separately and in interaction with the other input parameters.

In practice, we are usually interested in the first-order sensitivity indices S_1, \dots, S_d , the total ones S_1^T, \dots, S_d^T and sometimes in the second-order ones S_{ij} , $1 \leq i < j \leq d$, where S_i , S_{ij} and S_i^T measure the output sensitivities due to the main effect f_i of X_i , to the interaction f_{ij} between X_i and X_j and to all the X_i contributions $(f_u)_{u \ni i}$ respectively. The model f is devoid of interactions if $\sum_{i=1}^d S_i \approx 1$.

Sobol' indices are widely used in GSA because of their efficiency and facility of interpretation. Indeed, for a given input parameter, the Sobol' index measures its contribution to the mean squared deviation of the model output from its mean value. **However, from a physical point of view, engineers can be more interested in its contribution to the mean value of the output gradient.**

2.2 Derivative-based global sensitivity measures

Introduced by Sobol and Gresham (1995), sensitivity indices based on the partial derivatives of the computer code f have recently been studied by Sobol and Kucherenko (2009) and Lamboni et al. (2013), generalizing the importance measures introduced in the Morris method (Morris, 1991). They are based on the hypothesis that for a given input parameter x_k , $k \in \{1, \dots, d\}$, an important partial derivative with respect to this variable, over the whole probabilized input parameter space, leads to an important variation of the model output $f(x)$. These sensitivity indices are called "derivative-based global sensitivity measures" (DGSMs) and the k^{th} , k in $\{1, \dots, d\}$, is equal to:

$$D_k = \mathbb{E}_X \left[\left(\frac{\partial f(X)}{\partial x_k} \right)^2 \right] \quad (2)$$

with the notation $\mathbb{E}_X[\square] = \int_{\mathcal{X}} \square \mu_X(x)$ where μ_X , $\mu_X(x) > 0$ over \mathcal{X} , is the probability density function of X .

As mentioned by Touzani and Busby (2014), the DGSMs are tied with the Morris method which associates to each input parameter x_k the empirical mean $\mu_k = \frac{1}{R} \sum_{r=1}^R d_k^{(r)}$ and the standard deviation $\sigma_k = \sqrt{\frac{1}{R-1} \sum_{r=1}^R \left(d_k^{(r)} - \mu_k \right)^2}$ of the elementary effect $d_k^{(r)}$ defined as

$$d_k^{(r)} = \frac{f \left(X_1^{(r)}, \dots, X_{k-1}^{(r)}, X_k^{(r)} + \Delta, X_{k+1}^{(r)}, \dots, X_d^{(r)} \right) - f \left(X^{(r)} \right)}{\Delta}$$

with Δ the step of the input variation. First, we can rewrite the DGSM of x_k as $D_k = M_k^2 + \Sigma_k$ where $M_k = \mathbb{E}_X \left[\frac{\partial f(X)}{\partial x_k} \right]$ and $\Sigma_k = \mathbb{V}_X \left[\frac{\partial f(X)}{\partial x_k} \right]$. Then, we note that M_k and Σ_k generalize μ_k and σ_k^2 respectively: the first quantities consider Monte-Carlo approximations of the first centered moments of the k^{th} partial derivative while the second ones consider their empirical versions replacing the partial derivative by a finite difference with step Δ . Consequently, if the step Δ is greater than the characteristic dimension of the physical problem, the Morris method focus on the global variation of the model output whereas the DGSM approach focus on local changes averaged over the whole input parameter space according to a probability measure. Lastly, for some non-monotonic functions, the Morris index μ_k can be almost equal to zero even if x_k has an important influence on the model output; the DGSM D_k allows to detect this dependence, with the same idea than the sensitivity measure proposed by Campolongo et al. (2007) which replaces the elementary effect by its absolute value in the μ_k and σ_k formulations.

Otherwise, Sobol and Kucherenko (2009) and Lamboni et al. (2013) compare DGSMs and Sobol' indices from a theoretical point of view and show that for a large category of continuous functions, the total Sobol' index S_k^T is bounded by the DGSM index D_k weighted by a constant, function of the probability law of X_k . Moreover, Kucherenko et al. (2009) show that the estimation of a DGSM index often requires much less model evaluations than the estimation of a Sobol' one. From both considerations, DGSMs could be used in screening to exclude the non-significant input parameters from the model specification, in a more economical way. This selection of influential variables should be almost the same than the one obtained with the Sobol' indices, because of the inequality linking these sensitivity measures. Nevertheless, these studies mention that ranking the input parameters according to the upper bound values

does not lead necessarily to the same order than ranking these parameters according to the Sobol' indices.

Last but not least, the computer code does not always provide the adjoint code for the gradient evaluation. Moreover, when this gradient can be get, the number of available computer experiments is often too low and the integrals are badly estimated in Expression (2). Consequently in this paper, we propose to replace the computer code f by a surrogate model \hat{f} with a tractable gradient which is less accurate but faster in execution (Forrester et al., 2008), so allowing to have a much higher number of Monte-Carlo runs for the expectation estimation. In this paper, we focus on the Gaussian process metamodel (Rasmussen and Williams, 2005), in the same way as Marrel et al. (2009) for the estimation of the Sobol' indices. This surrogate model is often powerful for industrial studies and has a stochastic formulation which can usefully provide confidence intervals for the DGSM estimation.

2.3 Gaussian process metamodel

Let Ω be a sample space. We suppose that the computer code f can be modeled as an instance of the Gaussian process (GP) $Z : (\mathbb{R}^d, \Omega) \rightarrow \mathbb{R}$ defined by

$$Z(x; \omega) \sim \mathcal{GP} (h(x)^T \beta, \sigma^2 (r(x, x') + \tau^2 \delta_{xx'})) \quad (3)$$

with mean $h(x)^T \beta$ and covariance structure $\sigma^2 (r(x, x') + \tau^2 \delta_{xx'})$, β being a regression parameter vector and (σ^2, τ^2) variance parameters. The vector $h(x) = (h_1(x), \dots, h_p(x))^T$ is made of p basis functions, which are commonly monomials, e.g. $h(x) = (1 \ x_1 \ \dots \ x_d)^T \in \mathbb{R}^{d+1}$ with $p = d + 1$, and $r(x, x')$ is a kernel function parameterized by an hyperparameter vector $\theta \in \Theta$. The random field mentioned in (3) can be decomposed into the sum of a deterministic term $h(x)^T \beta$ and a stochastic one $Z_0(x; \omega)$, where $Z_0(x; \omega)$ is a centered GP with same covariance.

From the GP formulation, we associate to the n -sample $(X^{(i)}, Y^{(i)})_{1 \leq i \leq n}$ the matrices $H \in \mathcal{M}_{n,p}(\mathbb{R})$ and $R \in \mathcal{M}_n(\mathbb{R})$ defined by $H = (h_j(X^{(i)}))_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}}$ and $R = (r(X^{(i)}, X^{(j)} + \tau^2 \delta_{ij}))_{1 \leq i, j \leq n}$, as well as the n -dimensional vectors $k(x) = (r(x, X^{(1)}) \dots r(x, X^{(n)}))^T$ and $\mathbf{Y} = (Y^{(1)} \dots Y^{(n)})^T$. Note that R and k depend on the hyperparameters θ .

Then Z_C , the GP (3) conditioned by this n -sample and the parameters $(\beta, \sigma^2, \tau^2, \theta)$, is the new probability law for the computer code f :

$$Z_C(x; \omega) = \left[Z(x; \omega) \mid (X^{(i)}, Y^{(i)})_{1 \leq i \leq n}, \beta, \sigma^2, \tau^2, \theta \right] \sim \mathcal{GP}(\hat{f}(x), s^2(x, x')). \quad (4)$$

Predictor. As an instance of this posterior random field $Z_C(x; \omega)$, the response of the computer code f at a given location x is predicted by the conditional mean

$$\hat{f}(x) = h(x)^T \beta + k(x)^T R^{-1} (\mathbf{Y} - H \beta) = h(x)^T \beta + k(x)^T \gamma \quad (5)$$

with $\gamma = R^{-1} (\mathbf{Y} - H \beta)$. The conditional covariance $s^2(x, x')$ gives the quadratic risk $s^2(x) := s^2(x, x)$ of $\hat{f}(x)$ and it is defined by

$$s^2(x, x') = \sigma^2 (r(x, x') - k(x)^T R^{-1} k(x')). \quad (6)$$

Estimation. Formula (5) and (6) can not be used directly because the parameters $(\beta, \sigma^2, \theta)$ have to be estimated, classically by cross-validation or likelihood maximization procedures. In this paper, we consider the second method; precisely, we iterate explicit computation steps, $\hat{\beta} = (H^T R^{-1} H)^{-1} H^T R^{-1} \mathbf{Y}$ and $\widehat{\sigma^2} = n^{-1} (\mathbf{Y} - H \hat{\beta})^T R^{-1} (\mathbf{Y} - H \hat{\beta})$, and a numerical minimization one, $\hat{\theta} \in \operatorname{argmin}_{\theta \in \Theta} \left\{ \widehat{\sigma^2} \sqrt{\det(R)} \right\}$. Then, the parameter set $(\beta, \theta, \sigma^2)$ in Equations (5) and (6) is replaced by $(\hat{\beta}, \hat{\theta}, \widehat{\sigma^2})$. Hereafter, we always consider the estimated parameters but write $(\beta, \theta, \sigma^2)$ in order to lighten up the equations.

Nugget effect. When the parameter τ^2 is forced to zero, the predictor $\hat{f}(x)$ interpolates the learning sample $(X^{(i)}, Y^{(i)})_{1 \leq i \leq n}$ (Rasmussen and Williams, 2005). This situation is too restrictive for the predictor shape, e.g. an important value of n can provide an ill-conditioned correlation matrix R and a sharp predictor. On the contrary, allowing a non-null value for τ^2 often leads to a smoother predictor $\hat{f}(x)$ and a higher generalization ability; this change is called a nugget effect. The value of τ^2 can be optimized by

maximum likelihood, jointly with the other hyperparameters.

Covariance choice. Finally, we limit ourselves to the case where the kernel $r(x, x')$ is a tensorized product of monodimensional stationary kernels: $r(x, x') = \prod_{i=1}^d r_i(x_i - x'_i)$. This common hypothesis allows the mathematical developments presented in this paper. Furthermore, we suppose that these monodimensional kernels come from a same family of covariance functions that the user has to choose: exponential, Gaussian, Matérn, ... Complementary informations about Gaussian process metamodels can be found in the book *Gaussian Processes for Machine Learning* of Rasmussen and Williams (2005).

2.4 GPM-based estimation for DGSMs

In this paper, we propose two approaches for the estimation of the DGSM D_k , $k \in \{1, \dots, d\}$, using a GPM in the same way as Marrel et al. (2009). The first one uses no more than the estimator $\hat{f}(x)$ of $f(x)$ defined by (5) and builds the plug-in estimator

$$\hat{D}_k^{(1)} = \mathbb{E}_X \left[\left(\frac{\partial \hat{f}(X)}{\partial x_k} \right)^2 \right].$$

The second approach uses the law of the stochastic DGSM $\widehat{RD}_k^{(2)}(\omega)$ defined by the random variable:

$$\widehat{RD}_k^{(2)}(\omega) = \mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right]. \quad (7)$$

From this, the DGSM D_k is estimated by the full-GPM estimator which is the $\hat{D}_k^{(2)}(\omega)$ expectation:

$$\hat{D}_k^{(2)} = \mathbb{E}_\omega \left[\widehat{RD}_k^{(2)}(\omega) \right].$$

Similarly to the use of $s^2(x) = \mathbb{V}_\omega [Z_C(x, \omega)]$ as an indicator of the $\hat{f}(x)$ error, we could quantify the $\hat{D}_k^{(2)}$ estimation error using the $\widehat{RD}_k^{(2)}(\omega)$ variance:

$$\hat{s}_k^2 = \mathbb{V}_\omega \left[\widehat{RD}_k^{(2)}(\omega) \right]. \quad (8)$$

Confidence intervals associated to $\hat{D}_k^{(2)}$ can also be computed using intensive Monte-Carlo sampling of the random variable $\widehat{RD}_k^{(2)}(\omega)$, as developed in Section 3.

2.4.1 Plug-in estimation of a DGSM

The plug-in estimation of the DGSM consists in the computation of $\hat{D}_k^{(1)}$, which corresponds to the DGSM directly estimated on the GPM predictor \hat{f} given in Equation (5). Knowing that $\hat{f}(x) = h(x)^T \beta + k(x)^T \gamma$, the k^{th} partial derivative of $\hat{f}(x)$ is

$$\frac{\partial \hat{f}(x)}{\partial x_k} = \frac{\partial h(x)^T}{\partial x_k} \beta + \frac{\partial k(x)^T}{\partial x_k} \gamma$$

where $\frac{\partial h(x)}{\partial x_k} = \left(\frac{\partial h_1(x)}{\partial x_k}, \dots, \frac{\partial h_p(x)}{\partial x_k} \right)^T$ and $\frac{\partial k(x)}{\partial x_k} = \left(\frac{\partial r(x, X^{(1)})}{\partial x_k}, \dots, \frac{\partial r(x, X^{(n)})}{\partial x_k} \right)^T$.

Under the hypothesis of a tensorized product of monodimensional kernels for the covariance (see Section 2.3), we obtain for any $i \in \{1, \dots, n\}$:

$$\frac{\partial r(x, X^{(i)})}{\partial x_k} = \frac{\partial r_k(x_k, X_k^{(i)})}{\partial x_k} \prod_{\substack{j=1 \\ j \neq k}}^d r_j(x_j, X_j^{(i)}).$$

Then, considering element-wise mean for matrices, i.e. $(\mathbb{E}[A])_{ij} = \mathbb{E}[A_{ij}]$ for any real matrix A , we obtain a first estimator of the sensitivity measure D_k :

$$\hat{D}_k^{(1)} = \beta^T \mathbb{E}_X \left[\frac{\partial h(X)}{\partial x_k} \frac{\partial h(X)^T}{\partial x_k} \right] \beta + \gamma^T \mathbb{E}_X \left[\frac{\partial k(X)}{\partial x_k} \frac{\partial k(X)^T}{\partial x_k} \right] \gamma + 2\beta^T \mathbb{E}_X \left[\frac{\partial h(X)}{\partial x_k} \frac{\partial k(X)^T}{\partial x_k} \right] \gamma \quad (9)$$

whose integrals have to be analytically or numerically computed.

In this paper, we consider the most common situation where the prior mean is constant; consequently, we have $h(x) = 1$ and Equation (9) becomes

$$\hat{D}_k^{(1)} = \gamma^T \mathbb{E}_X \left[\frac{\partial k(X)}{\partial x_k} \frac{\partial k(X)^T}{\partial x_k} \right] \gamma.$$

Remark. Extensions of the following results to a more complex prior mean, such as a polynomial, are straightforward from Equation (9).

Then, thanks to the tensorized form of the kernel functions and of the independence hypothesis of the input parameters X_1, \dots, X_d , the matrix $A^{[k]} = \mathbb{E}_X \left[\frac{\partial k(X)}{\partial x_k} \frac{\partial k(X)^T}{\partial x_k} \right]$ can be better rewritten

$$A^{[k]} = \mathbb{E}_{X_k} \underbrace{\left[\frac{\partial k_k(X_k)}{\partial x_k} \frac{\partial k_k(X_k)^T}{\partial x_k} \right]}_{B^{[k]}} \bigodot_{\substack{l=1 \\ l \neq k}}^d \underbrace{\mathbb{E}_{X_l} [k_l(X_l) k_l(X_l)^T]}_{C^{[l]}} \quad (10)$$

where $k_l(x) = \left(r_l(x_l, X_l^{(1)}) \dots r_l(x_l, X_l^{(n)}) \right)^T$, for any $l \in \{1, \dots, d\}$ and \bigodot is the element-wise multiplication operator. The computation of $A^{[1]}, \dots, A^{[d]}$ requires $n(n+1)d$ monodimensional integral evaluations, because these matrices are function of the symmetric matrices $B^{[1]}, C^{[1]}, \dots, B^{[d]}, C^{[d]}$, each of which having $\frac{n(n+1)}{2}$ different elements of the form

$$C_{ij}^{[k]} = \mathbb{E}_{X_k} \left[r_k(X_k, X_k^{(i)}) r_k(X_k, X_k^{(j)}) \right], \quad (11)$$

or

$$B_{ij}^{[k]} = \mathbb{E}_{X_k} \left[\frac{\partial r_k(X_k, X_k^{(i)})}{\partial x_k} \frac{\partial r_k(X_k, X_k^{(j)})}{\partial x_k} \right]. \quad (12)$$

For particular cases of covariance functions and input parameter laws, analytical simplifications can be applied to Equation (10). More generally, these integrals can be quickly approached by Monte-Carlo sampling.

2.4.2 Full-GPM estimation of a DGSM

Another approach consists in the estimation of the sensitivity measure D_k based on the full Gaussian process $Z_C(x; \omega)$, rather than on its mean only. More precisely, we consider the full-GPM estimator $\hat{D}_k^{(2)}$ which can be easily developed as

$$\hat{D}_k^{(2)} = \hat{D}_k^{(1)} + \mathbb{E}_X \left[\mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right]. \quad (13)$$

In this way, the estimator $\hat{D}_k^{(2)}$ is equal to the plug-in one, $\hat{D}_k^{(1)}$, completed by an additive positive term associated to the variance of the GPM; consequently, we have $\hat{D}_k^{(2)} \geq \hat{D}_k^{(1)}$. We explicit this new part in the following manner:

$$\mathbb{E}_X \left[\mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right] = \sigma^2 \left\{ \mathbb{E}_X \left[\frac{\partial^2 r(U, V)}{\partial u_k \partial v_k} \Big|_{(U, V) := (X, X)} \right] - \mathbf{1}^T \left(R^{-1} \bigodot A^{[k]} \right) \mathbf{1} \right\} \quad (14)$$

where $\mathbf{1}$ is the all-ones element of \mathbb{R}^n . A new term appears in this expression,

$$\mathbb{E}_X \left[\frac{\partial^2 r(U, V; \theta)}{\partial u_k \partial v_k} \Big|_{(U, V) := (X, X)} \right], \quad (15)$$

which is easily estimable by Monte-Carlo sampling. Computational details about (13) and (14) are given in Appendix.

In addition, the full-GPM estimator (13) can be completed by an approximation of its error using the variance (8) or a confidence interval computed by simulations.

The next part of this paper deals with the computation of terms (11), (12) and (15) required for the approximation of $\hat{D}^{(1)}$ and $\hat{D}^{(2)}$ and for the construction of confidence intervals. Significance tests are also proposed.

3 Numerical implementation of the GPM-based DGSM estimators

In this part, we propose different numerical implementations of the DGSM estimators derived from GPM and defined in Section 2. Precisely, we compute the matrices (11), (12) and (15) used in $\hat{D}_k^{(1)}$ and $\hat{D}_k^{(2)}$, analytically or with a Monte-Carlo method.

3.1 Analytical results for uniform laws and Gaussian kernels

First, we consider a specific covariance function for the GP and a certain type of input distribution, in order to get analytical results for the computation of the estimator $\hat{D}_k^{(1)}$ of the sensitivity index D_k . More precisely, we consider classical Gaussian kernel functions

$$r_k(x_k, x'_k) = \exp\left(-\frac{(x_k - x'_k)^2}{2\theta_k^2}\right), \quad k \in \{1, \dots, d\},$$

and independent input parameters X_1, \dots, X_d following uniform laws:

$$X_k \sim \mathcal{U}([m_k, M_k]), \quad k \in \{1, \dots, d\}.$$

First of all, under these considerations and using Lemma 1 (see Appendix), we rewrite the term (11)

$$C_{ij}^{[k]} = \frac{\sqrt{\pi}\theta_k \mathbb{P}\left[W_{i,j}^{[k]} \in [m_k, M_k]\right] e^{-\frac{(X_k^{(i)} - X_k^{(j)})^2}{4\theta_k^2}}}{M_k - m_k} \quad (16)$$

where $W_{i,j}^{[k]} \sim \mathcal{N}\left(\frac{X_k^{(i)} + X_k^{(j)}}{2}, \frac{\theta_k^2}{2}\right)$.

Furthermore, using Lemma 2 (see Appendix), we can show that the term (12) is equal to

$$\begin{aligned} B_{ij}^{[k]} &= -\frac{(X_k^{(i)} - X_k^{(j)})^2}{4\theta_k^4} C_{ij}^{[k]} \\ &+ \frac{\sqrt{\pi}}{2\theta_k(M_k - m_k)} \left\{ a_{i,j}^{[k]} \varphi\left(a_{i,j}^{[k]}\right) - b_{i,j}^{[k]} \varphi\left(b_{i,j}^{[k]}\right) + \mathbb{P}\left[Z_{i,j}^{[k]} \in [a_{i,j}^{[k]}, b_{i,j}^{[k]}]\right] \right\} e^{-\frac{(X_k^{(i)} - X_k^{(j)})^2}{4\theta_k^2}} \end{aligned} \quad (17)$$

where $Z_{i,j}^{[k]} \sim \mathcal{N}(0, 1)$, $a_{i,j}^{[k]} = \frac{2m_k - X_k^{(i)} - X_k^{(j)}}{\sqrt{2\theta_k}}$, $b_{i,j}^{[k]} = \frac{2M_k - X_k^{(i)} - X_k^{(j)}}{\sqrt{2\theta_k}}$ and $\varphi(\cdot)$ is the probability density function of the standard normal law.

Finally, we have

$$\mathbb{E}_X \left[\frac{\partial^2 r(U, V; \theta)}{\partial u_k \partial v_k} \Big|_{(U, V) := (X, X)} \right] = \frac{1}{\theta_k^2}.$$

Expressions (16) and (17) only use evaluations of probability density and distribution functions. Consequently, the DGSM estimators $\hat{D}_k^{(1)}$ and $\hat{D}_k^{(2)}$ do not require integral quadratures to approach (11), (12) and (15) in presence of Gaussian kernel functions and uniform distributions for the input parameters. Under these assumptions, Section 4 illustrates the computational superiority of this method in comparison to Monte-Carlo approaches.

3.2 Monte-Carlo approximation for general case

In spite of the exact formulations of the GPM-based DGSM estimators, kernel functions are not always Gaussian and the input distributions are not always uniform in many industrial applications. Consequently, in such cases, numerical methods are required for the approximation of both DGSM estimators defined by (9) and (13).

The first estimator given in (9) is a plug-in one, which represents the mean of the squared estimator of a f partial derivative. This estimator having a deterministic and analytical formulation, the mean is approached by crude Monte-Carlo according to the specified input distributions; the computational cost is not high because the method only requires to estimate monodimensional integrals. A similar sampling technique can approach the variance (8) expressed in terms of simple and double integrals.

The second estimator given in (13) is more complex because it represents the stochastic mean of a squared Gaussian process averaged over the input parameter space \mathcal{X} :

$$\hat{D}_k^{(2)} = \mathbb{E}_\omega \left[\mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right] \right].$$

In order to get the second GPM-based estimator $\hat{D}_k^{(2)}$, we decide to approach numerically the distribution of the stochastic DGSM $\mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right]$ using N instances

$$\mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega_1)}{\partial x_k} \right)^2 \right], \dots, \mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega_N)}{\partial x_k} \right)^2 \right],$$

or more precisely using N instances

$$M^{-1} \sum_{i=1}^M \left(\frac{\partial Z_C(X^{(i)}; \omega_1)}{\partial x_k} \right)^2, \dots, M^{-1} \sum_{i=1}^M \left(\frac{\partial Z_C(X^{(i)}; \omega_N)}{\partial x_k} \right)^2$$

where $\mathbf{X} = (X^{(1)}, \dots, X^{(M)})$ are M i.i.d. instances of X . In this case, we have

$$\frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k} := \begin{pmatrix} \frac{\partial Z_C(X^{(1)}; \omega)}{\partial x_k} \\ \vdots \\ \frac{\partial Z_C(X^{(M)}; \omega)}{\partial x_k} \end{pmatrix} \sim \mathcal{N}(b_M, C_M)$$

with

$$(b_M)_i = \frac{\partial h(X^{(i)})^T}{\partial x_k} \beta + \frac{\partial k(X^{(i)})^T}{\partial x_k} \gamma, \quad i \in \{1, \dots, M\}$$

and

$$(C_M)_{ij} = \sigma^2 \left(\frac{\partial^2 r(U, V)}{\partial u_k \partial v_k} \Big|_{(U, V) = (X^{(i)}, X^{(j)})} - \frac{\partial k(X^{(i)})^T}{\partial x_k} R^{-1} \frac{\partial k(X^{(j)})}{\partial x_k} \right), \quad i, j \in \{1, \dots, M\}.$$

In the following, we propose two options for the simulation of instances of the random variable

$$M^{-1} \sum_{i=1}^M \left(\frac{\partial Z_C(X^{(i)}; \omega)}{\partial x_k} \right)^2, \quad (18)$$

in order to estimate its mean and, in a second phase, an associated confidence interval. The first one is based on a propagative version of the Gibbs sampler which approximates the law of the Gaussian vector $\frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k}$ and the second one is based on a chi-square approximation of the probabilistic law of the quadratic form (18).

3.2.1 Use of a propagative version of the Gibbs sampler

Concerning the first option, we simulate realizations of the Gaussian vector $\frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k}$ using the propagative version of the Gibbs sampler introduced by Lantuéjoul and Desassis (2012). More precisely, we consider

the blocking strategy of pivots presented in this paper and described in Algorithm 1. This propagative version is more efficient than the classical Gibbs sampler introduced by Geman and Geman (1984), because it does not require the inversion of the matrix C_M which is high-dimensional. The default value for the block length L is 5, which is used by Lantuéjoul and Desassis (2012).

With this method, it is possible to approach the distribution of the stochastic DGSM $\hat{D}_k^{(2)}(\omega)$. For this end, we define the i^{th} instance of the estimation of $\hat{D}_k^{(2)}$ based on Algorithm 1 by

$$\hat{D}_{k,i}^{(2),M} = \frac{1}{M} \sum_{l=1}^M \left(\frac{\partial Z_C(\mathbf{X}; \omega^{(i)})}{\partial x_k} \right)^2 \quad (19)$$

for all $i \in \{1, \dots, N\}$. Then, we propose to estimate $D^{(2)}$ by

$$\hat{D}_k^{(2),N,M} = \frac{1}{N} \sum_{i=1}^N \hat{D}_{k,i}^{(2),M} \quad (20)$$

and its confidence interval of level α by

$$\text{CI}_\alpha \left(D_k^{(2)} \right) = \left[q_{k,\alpha/2}^{(2),M,N}, q_{k,1-\alpha/2}^{(2),M,N} \right] \quad (21)$$

where $q_{k,\alpha/2}^{(2),M,N}$ and $q_{k,1-\alpha/2}^{(2),M,N}$ are the $\alpha/2$ and $1-\alpha/2$ empirical quantiles of the sample $\left(\hat{D}_{k,i}^{(2),M} \right)_{1 \leq i \leq N}$. In this paper, we compute the quantities using the `quantile` function of R with the default parametrization.

Data: the mean vector b_M , the covariance matrix C_M , the block length L and the number of instances N ;

Result: N realizations of the random vector $\frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k}$;

extract $s = \left(\frac{1}{(C_M)_{ii}} \right)_{1 \leq i \leq M}$;

update $C_M := C_M \odot (ss^T)$ where \odot is the element-wise multiplication operator;

let $A^{(0)} := 0_{\mathbb{R}^M}$;

for $i = 1, \dots, N$ **do**

1. take a sample $S_i = \{i_1, \dots, i_L\}$ of L integers in $\{1, \dots, M\}$ without replacement;
2. $A^{(i)} = A^{(i-1)} + C_{M,i} B_i^{-1} (U_i - A^{(i-1)}[S_i])$, where :
 - $U_i \sim \mathcal{N}(0_{\mathbb{R}^L}, I_L)$,
 - B_i is the upper triangular matrix of the $C_M[S_i, S_i]$ Cholesky decomposition,
 - $C_{M,i} := C_M[:, S_i]$.

end

for $i = 1, \dots, N$ **do**

1. $\frac{\partial Z_C(\mathbf{X}; \omega^{(i)})}{\partial x_k} := b_M + A^{(i)} \odot s$.

end

Algorithm 1: Propagative version of the Gibbs sampler with the blocking strategy of pivots.

Remark 1 This method based on the Gibbs sampler can be easily extended to the crossed DGSM introduced by Roustant et al. (2014):

$$\hat{D}_{ij} = \mathbb{E}_\omega \left[\mathbb{E}_X \left[\left(\frac{\partial^2 Z_C(X; \omega)}{\partial x_i \partial x_j} \right)^2 \right] \right].$$

Such a sensitivity measure allows to quantify the influence of an interaction between two input parameters on the output. Roustant et al. (2014) recommend their use for the detection of additive structure in the model; indeed, the authors prove that a crossed DGSM D_{ij} equal to zero implies that the input parameters x_i and x_j do not interact together in the model.

3.2.2 Use of a chi-square approximation

Concerning the second option, we consider a chi-square approximation of the stochastic DGSM $\hat{D}_k^{(2)}(\omega)$:

$$\begin{aligned} \widehat{RD}_k^{(2)}(\omega) \approx \hat{D}_k^{(2),M}(\omega) &= \frac{1}{M} \sum_{i=1}^M \left(\frac{\partial Z_C(X^{(i)}; \omega)}{\partial x_k} \right)^2 \\ &= \left(\frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k} \right)^T A \frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k} =: Q_k(\omega; \mathbf{X}) \end{aligned} \quad (22)$$

where $A = MI_M \in \mathcal{M}_M(\mathbb{R})$. Recently, Liu et al. (2009) propose to approach the distribution of the quadratic form $Q_k(\omega; \mathbf{X})$ using a noncentral chi-squared distribution $\chi_l^2(\delta)$ with $(l, \delta) \in \mathbb{N}^* \times \mathbb{R}_+$. This method does not require any matrix inversion or spectral decomposition, which is an advantage when the dimension of the covariance matrix C_M is important. More precisely, the authors rewrite $Q_k(\omega; \mathbf{X})$ as a weighted sum of non-central chi-squared variables whose weights are the eigenvalues of $C_M^{1/2} A C_M^{1/2}$. Then, they determine the hyperparameters of the $\chi_l^2(\delta)$ distribution so that its skewness is equal to the $Q_k(\omega; \mathbf{X})$ one and the distance between the kurtoses of these random variables is minimal.

From this chi-square approximation of the $Q_k(\omega; \mathbf{X})$ distribution, it is possible to approach quickly the mean of $\mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right]$ and its quantiles, in order to obtain an estimator $\hat{D}_k^{(2)}$ of D_k with a confidence interval. This method is described in Algorithm 2.

Data: the mean vector b_M , the covariance matrix C_M and the number of instances N ;

Result: N realizations of the random variable $\hat{D}_k^{(2),M}(\omega)$;

$$c_1 = n^{-1} \text{Tr}(C_M) + n^{-1} b_M^T b_M;$$

$$c_2 = n^{-2} \text{Tr}(C_M^2) + 2n^{-2} b_M^T C_M b_M;$$

$$c_3 = n^{-3} \mathbf{1}^T (C_M^2 \odot C_M) \mathbf{1} + 3n^{-3} (C_M b_M)^T C_M b_M;$$

$$c_4 = n^{-4} \mathbf{1}^T (C_M^2 \odot C_M^2) \mathbf{1} + 4n^{-4} \left((C_M^T)^T b_M \right)^T C_M b_M;$$

$$s_1 = c_3 c_2^{-1.5}, s_2 = c_4 c_2^{-2}, \mu_1 = c_1 \text{ and } \sigma_1 = \sqrt{2c_2};$$

if $s_1^2 > s_2$ **then**

$$a = (s_1 - \sqrt{s_1^2 - s_2})^{-1};$$

$$\delta = s_1 a^{-3} - a^{-2};$$

$$l = a^{-2} - 2\delta;$$

else

$$a = s_1^{-1};$$

$$\delta = 0;$$

$$l = c_3^3 c_3^{-2};$$

end

$$\mu_2 = l + \delta;$$

$$\sigma_2 = \sqrt{2a};$$

for $i = 1, \dots, N$ **do**

$$\hat{D}_{k,i}^{(2),M} = (u_i - \mu_2) \sigma_2^{-1} \sigma_1 + \mu_1 \text{ where } u_i \sim \chi_l^2(\delta);$$

end

Algorithm 2: Stochastic DGSM sampling using a noncentral chi-squared approximation.

At the end of Algorithm 2, we can compute the estimator $\hat{D}_k^{(2),N,M}$ and the confidence interval $\text{CI}_\alpha \left(D_k^{(2)} \right)$ given in Equations (20) and (21).

In Section 4, the different Monte-Carlo approaches are compared from accuracy and CPU time points of view.

3.3 Significance tests

Finally, we would like to use the GPM-based DGSM estimators for a screening purpose, in order to distinguish the non-significant input parameters and the significant ones. To this end, we propose to test the nullity of the DGSM D_k using a statistical significance test. In this case, the null hypothesis is “ $\mathcal{H}_0: D_k = 0$ ” and the alternative one is “ $\mathcal{H}_1: D_k \neq 0$ ”. These hypotheses can be rewritten “ $\mathcal{H}_0: \forall x \in \mathcal{X}, \frac{\partial f(x)}{\partial x_k} = 0$ ” and “ $\mathcal{H}_1: \exists \mathcal{X}_0 \subset \mathcal{X}, \forall x \in \mathcal{X}_0, \frac{\partial f(x)}{\partial x_k} \neq 0$ ”.

Under \mathcal{H}_0 , the mean m_M of the Gaussian vector $\frac{\partial Z_G(\mathbf{X})}{\partial x_k}$ is the null vector of \mathbb{R}^M . Consequently, the p -value associated to this statistical test and to the estimator $\hat{D}_k^{(2)}$ is

$$p_{\text{val},k} = \mathbb{P} \left[Q_k(\omega; \mathbf{X}) > \hat{D}_k^{(2)} | \mathcal{H}_0 \right],$$

and we use the chi-square approximation or the Gibbs sampler propagative version in order to approximate this probability under \mathcal{H}_0 . In this way, we conclude that X_k is significant if $p_{\text{val},k}$ is lower than some level α , usually equal to 5% or 10%. Such statistical tests are useful for a quick screening step.

4 Numerical experiments

In this section, we propose some numerical experiments to compare the different GPM-based DGSM estimators previously developed. In a first part, we study the convergence of the plug-in and full-GPM DGSM estimators and compare, for the second one, the different Monte-Carlo approximations proposed in Section 3.2. In a second part, we apply the DGSMs to an industrial application and compare the results with those obtained with the classical Sobol’ indices; we also apply the significance test proposed in Section 3.3 for a screening purpose.

4.1 Comparison of the different estimators and numerical methods for the DGSM computation

In this first part, we are interested in the convergence of the different DGSM estimators (plug-in and full-DGSM) according to the learning sample size n and, for the full-GPM, in the comparison of the different Monte-Carlo approximation methods. We consider the analytical Ishigami function defined by

$$f(X) = \sin(X_1) + 7 \sin^2(X_2) + 0.1 X_3^4 \sin(X_1)$$

where X_1, X_2, X_3 are i.i.d. $\mathcal{U}([-\pi, \pi])$. Its partial derivatives are:

$$\frac{\partial f(X)}{\partial x_1} = \cos(X_1) (1 + 0.4 X_3^4), \quad \frac{\partial f(X)}{\partial x_2} = 14 \cos(X_2) \sin(X_2) \quad \text{and} \quad \frac{\partial f(X)}{\partial x_3} = 0.4 X_3^3 \sin(X_1).$$

Table 1 contains the theoretical DGSMs and Sobol’ indices; the conclusions obtained with the total Sobol’ indices and the DGSMs are qualitatively and quantitatively different. According to the first ones, the more influential input is X_1 (45% of the explained output variance) while X_2 is the more significant according to the second ones (57% of the DGSM sum). Concerning X_3 , results are roughly the same with both types of sensitivity indices.

	X_1	X_2	X_3
1st order Sobol	0.31	0.44	0
Total Sobol	0.56 (45%)	0.44 (36%)	0.24 (19%)
DGSM	7.7 (18%)	24.5 (57%)	11.0 (25%)

Table 1: Theoretical DGSMs and Sobol’ indices for the Ishigami function.

Procedure

In the following, the DGSM estimators are built from a GPM with a constant mean, a nugget effect and a tensorized covariance based on Gaussian kernels (see Section 2.3). First, we consider a n -sample $(X^{(i)}, f(X^{(i)}))_{1 \leq i \leq n}$ where $(X^{(i)})_{1 \leq i \leq n}$ is an optimized Latin Hypercube Sample (McKay et al., 1979) of size n in $[-\pi, \pi]^3$. This design of experiments is built using the `geneticLHS` function of the R package

lhs which is based on the S optimality criterion (Stein, 1987) and a genetic type algorithm. Then, the GPM is obtained using this learning sample and the R package `DiceKriging v 1.5.3` (Roustant et al., 2012), the parameters being estimated by likelihood maximization with a BFGS algorithm (Byrd et al., 1995).

From this GPM, we want to:

1. compare the plug-in and full-GPM DGSM estimators using the analytical results presented in Section 3.1;
2. compare the Monte-Carlo approximation methods from accuracy and CPU time points of view.

4.1.1 Comparison of plug-in and full-GPM DGSM estimators

First of all, we run both estimators for different learning sample sizes $n \in \{50, 100, 150, 200\}$ and, for each learning sample, we measure the accuracy of the surrogate model using the predictivity coefficient

$$Q^2 = 1 - \frac{\sum_{i=1}^{n_{\text{test}}} \left(Y^{\text{test},(i)} - \hat{Y}^{\text{test},(i)} \right)^2}{\sum_{i=1}^{n_{\text{test}}} \left(Y^{\text{test},(i)} - \bar{Y}^{\text{test}} \right)^2}$$

where $(X^{\text{test},(i)}, Y^{\text{test},(i)})_{1 \leq i \leq n_{\text{test}}}$ is a test sample built from an uniform grid of $n_{\text{test}} = 10000$ nodes and $\bar{Y}^{\text{test}} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} Y^{\text{test},(i)}$. Note that this accuracy criterion is also computed for the partial derivatives of the Ishigami function. Finally, all these steps are repeated 1000 times and we look at the mean, the 0.025 and the 0.975 quantiles of the Monte-Carlo distributions of both estimators.

Figure 1 represents the evolution of the plug-in and full-GPM DGSM estimators in function of the sample size n . It shows also the evolution of the full-GPM ones in function of the accuracy criterion Q^2 associated to the corresponding f partial derivatives. In this case, each star represents a full-GPM DGSM estimation associated to one sample size and one Monte-Carlo run. First of all, we can arrange the input parameters according to their influences almost surely, when the number of observations is greater than 170. Moreover, we can conclude that the second variable is the most influential from 50 observations. Thirdly, looking at the dashed lines which represent 0.025 and the 0.975 quantiles of the Monte-Carlo estimator distribution, the full-GPM DGSM estimators lead to possibly much higher values than the plug-in ones. This phenomenon can be explained by the difference between both estimators, which is positive and equal to the mean variance of the stochastic squared partial derivative of the conditioned Gaussian process (see Equation (13)). Indeed, this quantity can be important for small designs of experiments and tends to zero when the number of observations increases. The same phenomenon has been observed with a Matérn kernel.

Then, Figure 2 shows that the differences of convergence rate between the explanatory variable DGSMs is due to the fitting quality of the different partial derivatives which differs from an input parameter to another. Particularly, the performance of the GPM to predict the partial derivatives is weaker for the input parameter X_3 than for the other ones. This may be due to the presence of this variable only in an interaction term, with a small coefficient and under the form of a non-linear effect almost equal to zero over the half of the variation domain of X_3 . In this way, its influence could be hidden by the other input parameters for weak sample size.

Finally, both estimators converge to the same value which is the theoretical DGSM and for small sample sizes, the full-GPM one can make some important excesses (here with the input X_1) or get closer to the right value (here with the input X_2). Even if these numerical tests do not illustrate the compensation of the surrogate model error with the plug-in estimator, in absolute terms **the full-GPM estimator is more justified because it takes up the hypothesis underlying the GPM: “the model $f(x)$ is an instance $Z_C(x; \omega^*)$ of the Gaussian process (4)”**. Consequently, **under this assumption, the theoretical DGSM D_k is the DGSM of $Z_C(x; \omega^*)$, with an unknown $\omega^* \in \Omega$, and the best estimator in a mean-square sense is the mean of $\hat{D}_k^{(2)}(\omega)$ over Ω .**

4.1.2 Comparison of Monte-Carlo approximations for the stochastic DGSM

Now, we focus on the distribution of the random variable $\hat{D}^{(2)}(\omega)$ and compare the three following Monte-Carlo methods:

1. sampling according to the multivariate Gaussian law of $\frac{\partial Z_C(\mathbf{X}; \omega)}{\partial x_k}$, using the `mvrnorm` function of the R package `MASS`;

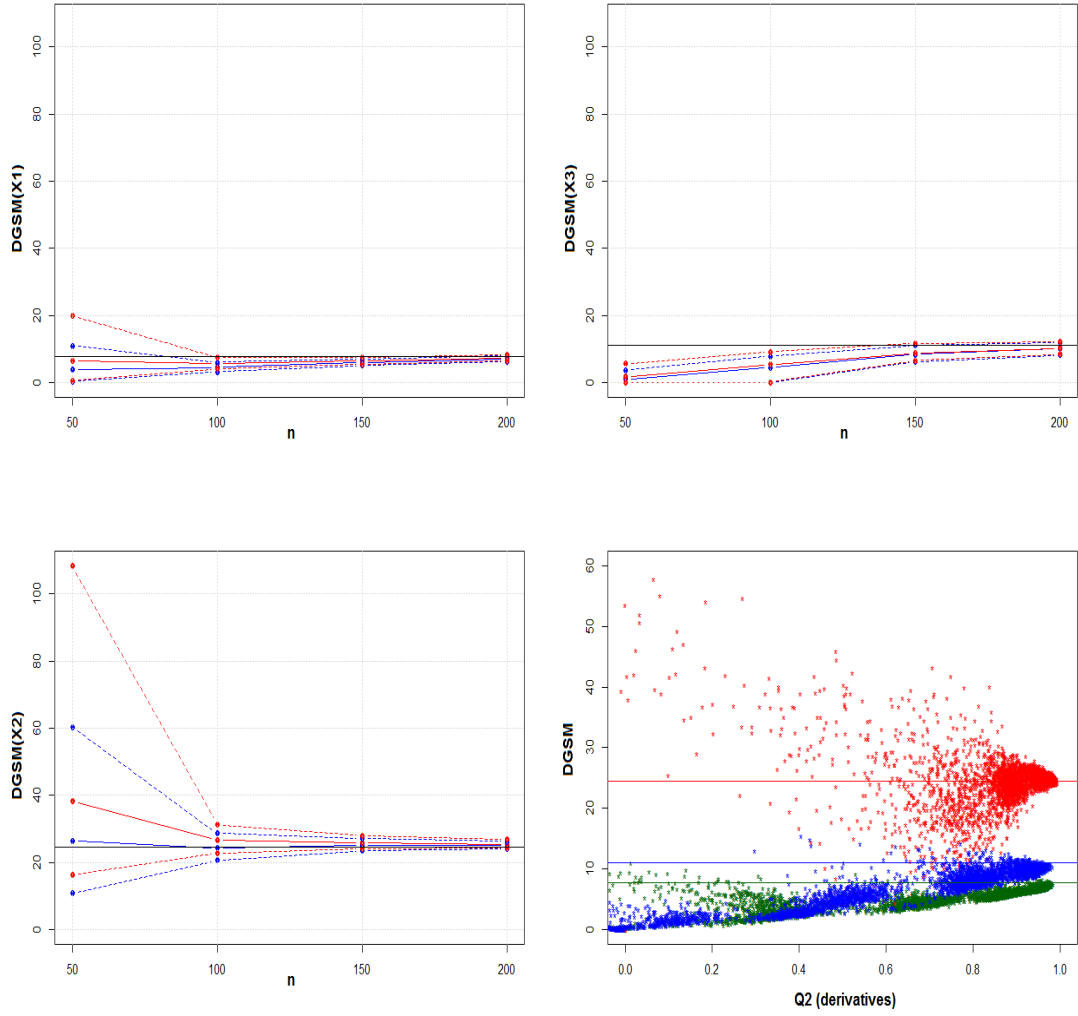


Figure 1: Evolution of the plug-in (blue) and full-GPM (red) DGSM estimators associated to the Ishigami function, in function of the number of observations n (three first figures). The solid lines represent the mean while the dashed ones represent the 2.5% and 97.5% quantiles of the Monte-Carlo distribution. The last picture represents the plug-in DGSM estimations associated to X_1 (green), X_2 (red) and X_3 (blue) in function of the accuracy criterion Q^2 of their associated partial derivatives.

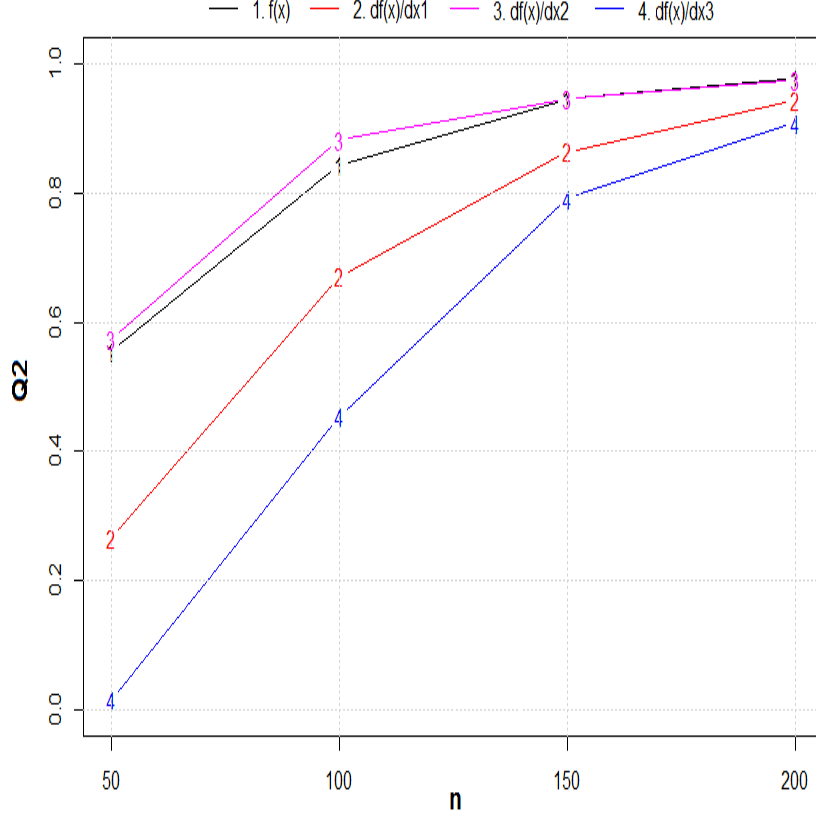


Figure 2: Evolution of the fitting accuracy of the Ishigami function and its partial derivatives according to the number of observations.

2. sampling using the propagative version of the Gibbs sampler based on a blocking strategy of pivots, proposed in Section 3.2.1;
3. sampling using the chi-square approximation proposed in Section 3.2.2.

The first method, based on a eigendecomposition of the covariance matrix, is used for reference. Indeed, from an accuracy point of view, this Monte-Carlo approach does not use any probability law approximation, contrary to the other ones. But this method is much more expensive from a CPU time point of view and would not be used in practice afterwards.

In this study, for different sample sizes n , we compute N instances of the stochastic DGSM $\hat{D}_3^{(2)}(\omega)$ associated to X_3 according to the three Monte-Carlo methods and using $M = 10000$ simulations in the $(\hat{D}_{3,i}^{(2),M})_{1 \leq i \leq N}$ computation (see Equation (19)), with $n \in \{50, 100, 200\}$ and $N \in \{500, 1000, 5000, 10000\}$. From these instances, we obtain the full-GPM estimator $\hat{D}_3^{(2)}$ and the 2.5% and 97.5% quantiles of the $\hat{D}_3^{(2)}(\omega)$ law.

Method	$N = 500$				$N = 1000$				$N = 5000$				$N = 10000$			
	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
$n = 50$	0.0123	5.9	0.9	2.1	0.0124	6.8	1.3	2.0	0.0118	14.4	4.6	1.9	0.0121	24.1	8.7	1.9
$n = 100$	0.0361	6.1	1.1	2.2	0.0339	6.6	1.4	2.0	0.0339	14.5	4.7	2.0	0.0344	24.2	8.7	2.0
$n = 200$	0.1343	6.3	1.3	2.5	0.1280	6.8	1.7	2.3	0.1284	14.8	5.0	2.3	0.130	24.5	9.0	2.3

Table 2: Comparison of the mean computational time (in seconds) for the Monte-Carlo approximation of the DGSM probabilistic laws using the `mvrnorm` function (1), the Gibbs sampler (2) or the chi-square approximation (3); the method indexed by 0 is the analytical one for uniform input laws and Gaussian kernels.

After $R = 100$ repetitions of the previous steps, Table 2 presents the mean computational time associated to the three Monte-Carlo methods, for different sample sizes n and different Monte-Carlo lengths N . **Firstly, the CPU time is almost insensitive to n . Secondly, when N is small with respect to M , the propagative version of the Gibbs sampler is faster than the chi-square approximation. To the contrary, when N is of the same order as M , this second approach is quicker than the first one. More generally, the chi-square approximation is all the more rapid than M is small with respect to N .** In the specific case where the estimator is the only quantity of interest, that is to say when standard deviation or confidence interval are not required, the use of analytical formula is very advised for uniform input laws and Gaussian kernels, with a CPU time between 10 and 100 times lower than the best one obtained with Monte-Carlo methods.

From an accuracy point of view, the differences between the different estimator errors are negligible with respect to the surrogate model error and the choice of a particular Monte-Carlo method has to be made according to CPU time considerations rather than accuracy ones.

To conclude, the propagative version of the Gibbs sampler and the chi-square approximations presented in Sections 3.2.1 and 3.2.2 are accurate approximations of the reference method using `mvrnorm`. From a CPU time point of view, we advise the use of the the Gibbs sampler when N is small with respect to M , and otherwise the use of the chi-square approximation.

4.2 Application to an environmental problem

Finally, we apply the DGSMs defined by Expression (2) to an environmental problem developed by CEA (France) and Kurchatov Institute (Russia) and dealing with a radioactive waste temporary storage site close to Moscow. The main objective is to predict the transport of strontium 90 between 2002 and 2010, in order to determine the aquifer contamination. This transport in saturated porous media is modeled by the MARTHE computer code (Volkova et al., 2008). In particular, this model returns in output the strontium 90 concentration predicted for year 2010 in many piezometers located on the waster repository site and we focus on one of them in this study. This quantity is function of 20 uncertain input parameters representing the permeability of different geological layers, longitudinal dispersivity coefficients, transverse dispersivity coefficients, sorption coefficients, porosity and meteoric water infiltration intensities. Marrel et al. (2009) ran a global sensitivity analysis from the more influential input variables, estimating the Sobol' indices with a Gaussian process metamodel. In the same way, we estimate the DGSMs from a GPM and from the tools developed in this paper, considering the whole set of input parameters.

Procedure

Concretely, we build a GPM from the 300 observations used by Marrel et al. (2009) with a constant mean, a nugget effect and a tensorized covariance based on Gaussian kernels. Its accuracy is measured with the predictivity coefficient Q^2 estimated by leave-one-out cross-validation (Hastie et al., 2001). Because of the high number of input parameters, we build this surrogate model iteratively using the efficient methodology of Marrel et al. (2008).

For the global sensitivity analysis, 17 out of 20 input parameters follow uniform distributions and the 3 remaining ones follow Weibull distributions, as mentioned in Table 3. So, we can not use the analytical formula of the DGSM estimators dedicated to uniform laws. Instead, we estimate them using the chi-square approximation approach with $M = 1000$.

Remark 2 *Note that in the initial test case, some factors are dependent: $d1-dt1$, $d2-dt2$, $d2-dt3$ and $i1-i2-i3$; this is not a problem for the DGSMs because the definition of these sensitivity indices is not based on a independence assumption, contrarily to the Sobol' indices for example. Furthermore, the DGSM estimation using the chi-square approximation method can deal with dependent input parameters, contrarily to the analytical formula of the DGSM estimators. However the interpretation of the results is more complicated in presence of dependent input variables and is beyond the remit of this study. Consequently, we replace $(dt1, dt2, dt3)$ by the ratios $(dt1/d1, dt2/d2, dt3/d3)$, which are independent of the other input*

parameters, and we consider the variations of infiltration coefficients between consecutive layers rather than their values. These modifications are mentioned by a bold italic index in Table 3.

	Input parameters	Notation	Distribution type	Distribution parameters ¹
1	Hydraulic conductivity layer 1	per1	Uniform	1, 15
2	Hydraulic conductivity layer 2	per2	Uniform	5, 0
3	Hydraulic conductivity layer 3	per3	Uniform	1, 15
4	Hydraulic conductivity zone 1	perz1	Uniform	1, 15
5	Hydraulic conductivity zone 2	perz2	Uniform	1, 15
6	Hydraulic conductivity zone 3	perz3	Uniform	1, 15
7	Hydraulic conductivity zone 4	perz4	Uniform	1, 15
8	Longitudinal dispersivity layer 1	d1	Uniform	0.05, 2
9	Longitudinal dispersivity layer 2	d2	Uniform	0.05, 2
10	Longitudinal dispersivity layer 3	d3	Uniform	0.05, 2
	Transversal dispersivity layer 1	dt1	Uniform	0.01× d1, 0.1× d1
11	Ratio of transversal to longitudinal dispersivity layer 1	dt1/d1	Uniform	0.01, 0.1
	Transversal dispersivity layer 2	dt2	Uniform	0.01× d2, 0.1× d2
12	Ratio of transversal to longitudinal dispersivity layer 2	dt2/d2	Uniform	0.01, 0.1
	Transversal dispersivity layer 3	dt3	Uniform	0.01× d3, 0.1× d3
13	Ratio of transversal to longitudinal dispersivity layer 3	dt3/d3	Uniform	0.01, 0.1
14	Volumetric distribution coefficient 1.1	kd1	Weibull	1.1597, 19.9875
15	Volumetric distribution coefficient 1.2	kd2	Weibull	0.891597, 24.4455
16	Volumetric distribution coefficient 1.3	kd3	Weibull	1.27363, 22.4986
17	Porosity	poros	Uniform	0.3, 0.37
18	Infiltration type 1	i1	Uniform	0, 0.0001
	Infiltration type 2	i2	Uniform	i1, 0.01
19	Infiltration type 2 - type 1	δi2	Uniform	0, 0.0092
	Infiltration type 3	i3	Uniform	i2, 0.1
20	Infiltration type 3 - type 2	δi3	Uniform	0, 0.096

Table 3: Input parameter description for the Marthe test case.

Results

The GPM built with the learning sample has a $Q^2 = 0.97$ using a leave-one-out cross-validation. Then we compute $N = 10000$ instances of the stochastic DGSM $\widehat{RD}_k^{(2)}(\omega)$ to obtain the mean $\hat{D}_k^{(2)}$, which is the estimator of D_k , the 2.5% and 97.5% quantiles and the p -value associated to the significance test described in Section 3.3. These quantities are gathered in Table 4. On the one hand, a simple glance to the different DGSM estimators leads to the conclusion that the input parameter **i1** explains the output gradient in its entirety, and the 95% intervals support these conclusion. On the other hand, statistical tests on the nullity of these DGSMs with a level $\alpha = 5\%$ conclude that **i1** is one of the less influential parameters, while the more influential ones are **kd1**, **δi3**, **perz2**, **per3**, **d3**, **perz3** and **δi2**. Consequently, the DGSMs lead to totally different conclusions, depending on whether the DGSMs or their significativities are considered.

In the face of these considerations, we propose a new sensitivity measure based on the stochastic DGSM:

$$\hat{\Delta}_k(\omega) = \sigma_k^2 \mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right]$$

and associates the normalized version:

$$\hat{\hat{\Delta}}_k(\omega) = \frac{\hat{\Delta}_k(\omega)}{\sum_{i=1}^d \hat{\Delta}_i(\omega)}.$$

This approach is similar to the one discussed by Saltelli et al. (2000) in their introduction, considering a SA index equal to the output partial derivative weighted by the input standard deviation and normalized by the standard deviation of the output. The index $\hat{\Delta}_k(\omega)$ weights the square infinitesimal output variation due to the variable x_k and averaged on the whole input parameter space, by the variance σ_k^2 of X_k , which characterizes the input parameter space spread along the x_k direction. We call it “weighted DGSM”. In this way,

- a very important DGSM, due to a single strong local variation along the direction of x_k whose uncertainty is very small, can be transformed into a small weighted DGSM, **while**
- a very small DGSM, due to many medium local variations along the direction of x_k whose uncertainty is very important, can be transformed into a medium or high DGSM.

Such behaviors are more in agreement with global sensitivity measure expectations: a high value when the input parameter strongly impacts the model output on average and a small one for the inverse situation. Moreover, the comparison of weighted DGSMs is more reasonable than the comparison of DGSMs. Indeed, while the first sensitivity measures have the same dimension, the second indices have different ones, each of them corresponding to the associated input parameter one. As an intuitive justification, the variance term σ_k^2 makes sense in a output variance decomposition, using a linear expansion of the function $f(x)$ around x^* :

$$f(x) \approx f^*(x) = f(x^*) + \sum_{k=1}^d \left. \frac{\partial f(x)}{\partial x_k} \right|_{x^*} (x_k - x_k^*).$$

In this case, we obtain for independent input parameters:

$$\mathbb{V}[f^*(X)] = \sum_{k=1}^d \left(\left. \frac{\partial f(x)}{\partial x_k} \right|_{x^*} \right)^2 \mathbb{V}[X_k] = \sum_{k=1}^d \left(\left. \frac{\partial f(x)}{\partial x_k} \right|_{x^*} \right)^2 \sigma_k^2$$

and then, the expectation with respect to x^* leads to the mean variance decomposition:

$$\mathbb{E}_{X^*} [\mathbb{V}[f^*(X)]] = \sum_{k=1}^d \sigma_k^2 D_k = \sum_{k=1}^d \Delta_k.$$

The second part of Table 4 presents the results associated to the propose weighted DGSM. With this perspective, the variable **kd1** explains 88% of the global output uncertainty while the **δi3** contribution is equal to 7% and **per1** and **perz2** to around 2%. The explanation of the output uncertainty is radically different from the DGSM point of view where these four variables are totally hidden by the input parameter **i1**, especially for **kd1** which is the most significant input according to the weighted DGSMs. Moreover, **the conclusions of the global sensitivity analysis obtained with the weighted DGSMs are of the same order than those obtained with the Sobol’ indices. Lastly, these conclusions are also similar to the ones based on the p-values, what numerically validates the screening method proposed in Section 3.3.**

Lastly, Figure 3 compares the p-values computed with the different Monte-Carlo methods and we can see that despite of quantitative differences, screening conclusions are similar if we use a significance test with a level $\alpha = 10\%$ rather than 5%. Indeed, because of the law approximations and the chosen M and N values, the estimation of the distribution tails can be rough, in particularly for chi-square approximation whose associated significance test is too conservative. Consequently, for a more robust screening, we recommend against using a too small level for screening.

To conclude, in addition to illustrating the use of DGSM-based tests for screening, this industrial application led us to apply a new sensitivity measure derived from the classical DGSM: the weighted DGSM. From the previous results, the comparison of weighted DGSMs can be considered as a method from the global sensitivity analysis toolbox while classical DGSMs can be considered as tools from the local sensitivity analysis toolbox. A classical DGSM focus on the mean square value of some partial derivative, that is to say on a punctual information averaged over the whole probabilized input parameter space, while a weighted DGSM focus on this value balanced by the associated input parameter uncertainty. The comparison of weighted DGSMs makes sense because they have the same physical dimension (the squared output one), contrary to the classical DGSMs whose physical dimensions differ. For both kinds of sensitivity indices, significance tests lead to the same conclusions because the statistics only differ by a constant multiplicative term.

X_k	Stochastic DGSM $\widehat{RD}_k^{(2)}(\omega)$				Weighted DGSM		Sobol 1st order
	Mean $\hat{D}_k^{(2)}$	q(2.5%)	q(97.5%)	Screening using $p_{val,k}$	Mean $\hat{\Delta}_k$	Mean $\hat{\Delta}_k$	
kd1	2.2e-04	2.0e-04	2.4e-04	0.000 ***	5.4e-02	88%	0.75
$\delta i3$	5.6e+00	5.1e+00	6.2e+00	0.000 ***	4.1e-03	7%	0.16
perz2	5.8e-05	4.5e-05	7.3e-05	0.000 ***	9.3e-04	2%	0.03
d1	2.5e-04	1.8e-04	3.4e-04	0.006 **	7.7e-05	0%	0.03
kd2	4.2e-08	2.9e-08	6.2e-08	0.113	3.5e-05	0%	0.03
per1	7.2e-05	6.0e-05	8.5e-05	0.000 ***	1.2e-03	2%	0.06
perz1	5.1e-06	3.6e-06	7.1e-06	0.001 **	8.9e-05	0%	0.03
dt1/d1	1.2e-02	8.7e-02	1.7e-01	0.001 **	8.4e-05	0%	0.03
per3	1.0e-05	7.1e-06	1.4e-05	0.000 ***	1.8e-04	0%	0.03
d3	2.7e-04	1.9e-04	3.6e-04	0.000 ***	8.4e-05	0%	0.03
per2	3.7e-06	2.7e-06	4.9e-06	0.022 *	6.9e-05	0%	0.03
perz4	4.0e-06	2.9e-06	5.3e-06	0.040 *	6.8e-05	0%	0.03
perz3	5.3e-06	3.7e-06	7.3e-06	0.000 ***	8.2e-05	0%	0.03
i1	8.0e+04	5.9e+04	1.1e+05	0.040 *	6.6e-05	0%	0.03
dt2/d2	1.1e-01	7.8e-02	1.5e-01	0.011 *	7.5e-05	0%	0.03
$\delta i2$	1.0e+01	7.1e+00	1.5e+01	0.000 ***	8.5e-05	0%	0.03
kd3	1.1e-07	7.4e-08	1.6e-07	0.215	2.8e-05	0%	0.03
d2	2.2e-04	1.6e-04	3.0e-04	0.037 *	6.8e-05	0%	0.03
dt3/d3	1.0e-01	7.6e-02	1.4e-01	0.034 *	6.7e-05	0%	0.03
poros	1.7e-01	1.2e-01	2.3e-01	0.028 *	7.0e-05	0%	0.03

Table 4: DGSM statistics for the MARTHE test case, using the classical formulation and weighted versions.

5 Conclusion

This paper deals with the estimation of the derivative-based global sensitivity measures (DGSMs) associated to a numerical simulator in global sensitivity analysis (GSA) (Sobol and Kucherenko, 2009). This work falls within the scenario where this model does not provide the adjoint code to compute its output gradient. Under this consideration, we replaced this simulator by a surrogate model, more precisely by a Gaussian process metamodel (GPM) (Rasmussen and Williams, 2005). This idea has already been investigated with other surrogate models or sensitivity measures (Sudret, 2008; Marrel et al., 2009; Sudret and Mai, 2014).

We proposed two kinds of GPM-based DGSM estimators. A first one consists in substituting the numerical model by the GPM predictor; we called it “plug-in estimator”. A second one uses the full GPM and we showed that this DGSM estimator is equal to the plug-in one, completed by an additive variance term; we called it “full-GPM estimator”. This second estimator is the expectation of a stochastic DGSM where the numerical model is replaced by the whole GPM, not only its mean contrarily to the plug-in estimator; in this way, it takes into account the metamodel error, contrarily to the plug-in one. Consequently, we can also approach the distribution of this stochastic DGSM, build many statistics and more particularly provide a confidence interval for the DGSM estimator. Using this full-GPM estimator, we also proposed a significance test for eliminating the non-influential variables in a screening context. For all these reasons, the full-GPM is more relevant than the plug-in one, especially in presence of the small learning sample.

These estimators require many integral evaluations which can be computed by Monte-Carlo sampling. To this end, we proposed efficient methods based either on a propagative version of the Gibbs sampler, or on a chi-square approximation. We advise the first approach when the number of Monte-Carlo replications is small with respect to the number of GPM simulations required to the approximation of a stochastic DGSM instance, and conversely, and vice versa. Both methods can be used for the significance test computation. Furthermore, we showed that these plug-in and full-GPM estimators are analytically tractable when the input parameters are uniformly distributed and when the kernel functions of the

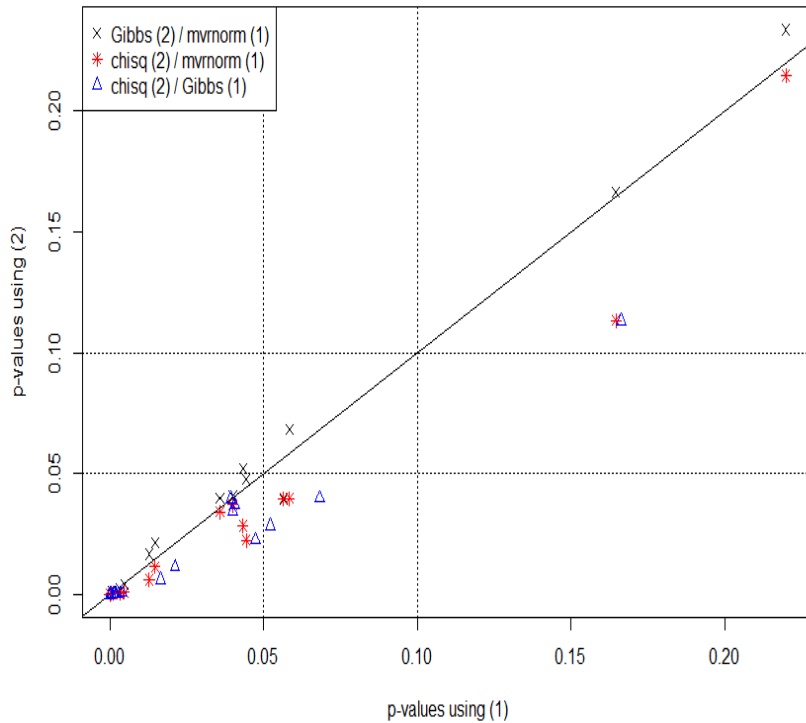


Figure 3: Comparison of the p-values computed by the different Monte-Carlo methods for the Marthe test case: reference method (`mvrnorm`), Gibbs sampler (`Gibbs`) and chi-squared approximation (`chisq`).

GPM are Gaussian, considerably reducing the computational time.

Finally, we ran numerical studies with the Ishigami function and consider an industrial application with a complex hydrogeological computer code. For the first point, the convergence rates of the DGSM estimators differ from an input parameter to another, which can be explained by the accuracy differences between the GPM partial derivatives. This is due to the fact that the GPM maximizes the likelihood on a set of computer code output observations, not on the corresponding partial derivative values. In other words, the metamodel is built without any idea of the behavior of the model gradient. However, we showed that, from a few observations, the estimators using the full-GPM can be used in a screening context to rank the input parameters according to their level of influence on the model output, and provide an idea of their relative significance. For the industrial application, the results give an input parameter order different than the one obtained with Sobol' indices (Marrel et al., 2009) but lead to the same conclusion about the influential variables using a significant test. This reinforces the pertinence of these DGSM estimators for screening purposes. Furthermore, we proposed a weighted version of the DGSM, after having established that the classical DGSMs do not take into account the uncertainty of their associated input parameters. To a certain extent, the classical DGSMs are useful for probabilized local sensitivity analysis while weighted DGSMs are useful for global sensitivity analysis.

To conclude, the GPM-based DGSM estimators developed in this paper are encouraging for local or global sensitivity analysis and for screening from a few evaluations of the numerical simulator. A criticism of this approach is that these estimators consider the gradient of the GPM which has been built using observations of the simulator output, not of its gradient. Consequently, nothing guarantees that the partial derivatives of the GPM are accurate approximations of the associated partial derivative of the simulator. For future research, adaptive sampling dedicated to the gradient approximation enhancement could be investigated: a strategy could be to add, as a new code simulation, the input parameter vector maximizing the trace of the GPM gradient covariance or the reduction of this trace. Moreover, the GPM-based DGSM estimations should be improved if some observations of the model output partial derivatives were available. In the case where the adjoint code is present, developments could be done about the integration of some model gradient simulations completing the output evaluations.

Acknowledgements

We are grateful to Béatrice Laurent for helpful discussions.

References

- Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM J. Scientific Computing*, 16:1190–1208.
- Campolongo, F., Cariboni, J., and Saltelli, A. (2007). An effective screening design for sensitivity analysis of large models. *Environmental Modelling & Software*, 22(10):1509 – 1518. Modelling, computer-assisted simulations, and mapping of dangerous phenomena for hazard assessment.
- Da Veiga, S. (2014). Global sensitivity analysis with dependence measures. *Journal of Statistical Computation and Simulation*.
- De Lozzo, M. and Marrel, A. (2015). New improvements in the use of dependence measures for sensitivity analysis and screening. Submitted, hal-01090475.
- de Rocquigny, E., Devictor, N., and Tarantola, S. (2008). *Uncertainty in industrial practice*. Wiley.
- Dreyfus, G. (2005). *Neural networks - methodology and applications*. Springer.
- Efron, B. and Stein, C. (1981). The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586–596.
- Forrester, A., Sobester, A., and Keane, A. (2008). *Engineering Design via Surrogate Modelling: A Practical Guide*. Wiley, 1 edition.
- Fort, J.-C., Klein, T., and Rachdi, N. (2013). New sensitivity analysis subordinated to a contrast. *Communications in Statistics - Theory and Methods*.
- Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-6(6):721–741.
- Gretton, G., Bousquet, O., Smola, A., and Schölkopf, B. (2005). Measuring statistical dependence with hilbert-schmidt norms. In *Proceedings Algorithmic Learning Theory*, pages 63–77. Springer-Verlag.
- Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.
- Homma, T. and Saltelli, A. (1996). Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1):1 – 17.
- Iooss, B. and Lemaître, P. (2015). A review on global sensitivity analysis methods. In Meloni, C. and Dellino, G., editors, *Uncertainty management in Simulation-Optimization of Complex Systems: Algorithms and Applications*. Springer.
- Iooss, B., Popelin, A.-L., Blatman, G., Ciric, C., Gamboa, F., Lacaze, S., and lamboni, M. (2012). Some new insights in derivative-based global sensitivity measures. In *Proceedings of SPAM11 & ESREL 2012 Conference*. ESRA.
- Kiparissides, A., Kucherenko, S. S., Mantalaris, A., and Pistikopoulos, E. N. (2009). Global Sensitivity Analysis Challenges in Biological Systems Modeling. *Industrial & Engineering Chemistry Research*, 48(15):7168–7180.
- Kleijnen, J. P. (1997). Sensitivity analysis and related analyses: A review of some statistical techniques. *Journal of Statistical Computation and Simulation*, 57(1-4):111–142.
- Kucherenko, S., Rodriguez-Fernandez, M., Pantelides, C., and Shah, N. (2009). Monte carlo evaluation of derivative-based global sensitivity measures. *Reliability Engineering & System Safety*, 94(7):1135 – 1148. Special Issue on Sensitivity Analysis.
- Lamboni, M., Iooss, B., Popelin, A.-L., and Gamboa, F. (2013). Derivative-based global sensitivity measures: General links with sobol’ indices and numerical tests. *Mathematics and Computers in Simulation*, 87:45–54.

- Lantuéjoul, L. and Desassis, N. (2012). Simulation of a gaussian random vector: A propagative version of the gibbs sampler. In *Ninth International Geostatistics Congress, June 11. - 15., 2012*. Oslo, Norway.
- Liu, H., Y., T., and Zhang, H. H. (2009). A new chi-square approximation to the distribution of non-negative definite quadratic forms in non-central normal variables. *Computational Statistics & Data Analysis*, 53(4):853 – 856.
- Marrel, A., Iooss, B., Laurent, B., and Roustant, O. (2009). Calculations of sobol indices for the gaussian process metamodel. *Reliability Engineering & System Safety*, 94(3):742–751.
- Marrel, A., Iooss, B., Van Dorpe, F., and Volkova, E. (2008). An efficient methodology for modeling complex computer codes with Gaussian processes. *Computational Statistics & Data Analysis*, 52:4731–4744.
- McKay, M. D., Beckman, R. J., and Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245.
- Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174.
- Plischke, E., Borgonovo, E., and Smith, C. L. (2013). Global sensitivity measures from given data. *European Journal of Operational Research*, 226(3):536 – 550.
- Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.
- Roustant, O., Fruth, J., Iooss, B., and Kuhnt, S. (2014). Crossed-derivative based sensitivity measures for interaction screening. *Mathematics and Computers in Simulation*, 105(0):105 – 118.
- Roustant, R., Ginsbourger, D., and Deville, Y. (2012). DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *Journal of Statistical Software*, 51(1):1–55.
- Saltelli, A., Chan, K., and Scott, E. M. (2000). *Sensitivity analysis*. Wiley, New York, NY.
- Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008). *Global sensitivity analysis: the primer*. John Wiley.
- Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). *Sensitivity analysis in practice: a guide to assessing scientific models*. John Wiley & Sons.
- Santner, T. J., Williams, B. J., and Notz, W. (2003). *The design and analysis of computer experiments*. Springer series in statistics. Springer, New York.
- Sobol, I. and Gresham, A. (1995). On an alternative global sensitivity estimators. In *Proceedings of SAMO 1995*, pages 40–42. Belgirate.
- Sobol, I. and Kucherenko, S. (2009). Derivative based global sensitivity measures and their link with global sensitivity indices. *Mathematics and Computers in Simulation*, 79(10):3009 – 3017.
- Sobol, I. M. (1993). Sensitivity estimates for nonlinear mathematical models. *MMCE*, 1:407–414.
- Soize, C. and Ghanem, R. (2004). Physical systems with random uncertainties: Chaos representations with arbitrary probability measure. *SIAM J. Scientific Computing*, 26(2):395–410.
- Stein, M. (1987). Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151.
- Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, 93(7):964 – 979. Bayesian Networks in Dependability.
- Sudret, B. and Mai, C. V. (2014). Computing derivative-based global sensitivity measures using polynomial chaos expansions. *Reliability Engineering & System Safety*.
- Székely, G. J., Rizzo, M. L., and Bakirov, N. K. (2007). Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794.

- Touzani, S. and Busby, D. (2014). Screening method using the derivative-based global sensitivity indices with application to reservoir simulator. *Oil Gas Sci. Technol. - Rev. IFP Energies nouvelles*.
- Volkova, E., Iooss, B., and Van Dorpe, F. (2008). Global sensitivity analysis for a numerical model of radionuclide migration from the rrc kurchatov institute radwaste disposal site. *Stochastic Environmental Research and Risk Assessment*, 22(1):17–31.

Mathematical details

Computation of $\hat{D}_k^{(2)}$

From the Gaussian process metamodel and the definition $\hat{D}_k^{(2)} = \mathbb{E}_\omega \left[\mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right] \right]$, we obtain the relation (13):

$$\begin{aligned}
\hat{D}_k^{(2)} &= \mathbb{E}_\omega \left[\mathbb{E}_X \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right] \right] = \mathbb{E}_X \left[\mathbb{E}_\omega \left[\left(\frac{\partial Z_C(X; \omega)}{\partial x_k} \right)^2 \right] \right] \\
&= \mathbb{E}_X \left[\left(\mathbb{E}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right)^2 + \mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right] \\
&= \mathbb{E}_X \left[\left(\frac{\partial \hat{f}(X)}{\partial x_k} \right)^2 \right] + \mathbb{E}_X \left[\mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right] \\
&= \hat{D}_k^{(1)} + \mathbb{E}_X \left[\mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right].
\end{aligned}$$

Then we develop the second term $\mathbb{E}_X \left[\mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right]$ and get the result (14):

$$\begin{aligned}
\mathbb{E}_X \left[\mathbb{V}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right] &= \mathbb{E}_X \left[\text{Cov}_\omega \left[\frac{\partial Z_C(X; \omega)}{\partial x_k}, \frac{\partial Z_C(X; \omega)}{\partial x_k} \right] \right] \\
&= \mathbb{E}_X \left[\frac{\partial^2}{\partial u_k \partial v_k} \text{Cov}_\omega [Z_C(U; \omega), Z_C(V; \omega)] \Big|_{(U, V) := (X, X)} \right] \\
&= \mathbb{E}_X \left[\frac{\partial^2}{\partial u_k \partial v_k} (\sigma^2 r(U, V) - \sigma^2 k(U)^T R^{-1} k(V)) \Big|_{(U, V) := (X, X)} \right] \\
&= \sigma^2 \mathbb{E}_X \left[\frac{\partial^2 r(U, V)}{\partial u_k \partial v_k} \Big|_{(U, V) := (X, X)} \right] - \sigma^2 \mathbb{E}_X \left[\frac{\partial k(U)^T}{\partial u_k} R^{-1} \frac{\partial k(V)}{\partial v_k} \Big|_{(U, V) := (X, X)} \right] \\
&= \sigma^2 \mathbb{E}_X \left[\frac{\partial^2 r(U, V)}{\partial u_k \partial v_k} \Big|_{(U, V) := (X, X)} \right] \\
&\quad - \sigma^2 \sum_{i, j=1}^n (R^{-1})_{ij} \mathbb{E}_X \left[\frac{\partial r(X, X^{(i)})}{\partial x_k} \frac{\partial r(X, X^{(j)})}{\partial x_k} \right] \\
&= \sigma^2 \mathbb{E}_X \left[\frac{\partial^2 r(U, V)}{\partial u_k \partial v_k} \Big|_{(U, V) := (X, X)} \right] \\
&\quad - \sigma^2 \sum_{i, j=1}^n (R^{-1})_{ij} \mathbb{E}_{X_k} \left[\frac{\partial r_k(X_k, X_k^{(i)})}{\partial x_k} \frac{\partial r_k(X_k, X_k^{(j)})}{\partial x_k} \right] \\
&\quad \times \prod_{\substack{l=1 \\ l \neq k}}^d \mathbb{E}_{X_l} [r_l(X_l, X_l^{(i)}) r_l(X_l, X_l^{(j)})].
\end{aligned}$$

Gaussian kernel and uniform laws

Lemma 1 Let $X \sim \mathcal{U}([m, M])$ and $r(x, x') = \exp\left(-\frac{(x-x')^2}{2\theta^2}\right)$. Then

$$\mathbb{E} [r(X, y)r(X, z)] = \sqrt{\pi}\theta(M - m)^{-1} \exp\left(-\frac{(y - z)^2}{4\theta^2}\right) \mathbb{P} [Z \in [m, M]]$$

where $Z \sim \mathcal{N}\left(\frac{y+z}{2}; \frac{\theta^2}{2}\right)$.

Proof 1

$$\begin{aligned}
\mathbb{E}[r(X, y)r(X, z)] &= \mathbb{E}\left[\exp\left(-\frac{2X^2 - 2X(y+z) + y^2 + z^2}{2\theta^2}\right)\right] \\
&= \mathbb{E}\left[\exp\left(-\frac{(X - \frac{y+z}{2})^2 + \frac{1}{4}(y-z)^2}{2(\frac{\theta}{\sqrt{2}})^2}\right)\right] \\
&= \sqrt{\pi}\theta \exp\left(-\frac{(y-z)^2}{4\theta^2}\right) \mathbb{E}\left[\frac{1}{\sqrt{2\pi}\frac{\theta}{\sqrt{2}}}\exp\left(-\frac{(X - \frac{y+z}{2})^2}{2(\frac{\theta}{\sqrt{2}})^2}\right)\right] \\
&= \sqrt{\pi}\theta \exp\left(-\frac{(y-z)^2}{4\theta^2}\right) (M-m)^{-1} \int_m^M \varphi_{\mathcal{N}(\frac{y+z}{2}, \frac{\theta^2}{2})}(x) dx
\end{aligned}$$

where $\varphi_{\mathcal{N}(m, \sigma^2)}(x)$ is the probability density function of the normal law with mean m and variance σ^2 .

Lemma 2 Let $X \sim \mathcal{U}([m, M])$ and $r(x, x') = \exp\left(-\frac{(x-x')^2}{2\theta^2}\right)$. Then

$$\begin{aligned}
\mathbb{E}\left[\frac{\partial r(X, y)}{\partial x} \frac{\partial r(X, z)}{\partial x}\right] &= -\frac{(y-z)^2}{4\theta^4} \mathbb{E}[r(X, y)r(X, z)] \\
&\quad + \frac{\sqrt{\pi}}{2\theta(M-m)} \exp\left(-\frac{(y-z)^2}{4\theta^2}\right) \{a\varphi(a) - b\varphi(b) - \mathbb{P}[Z \in [a, b]]\}
\end{aligned}$$

where $Z \sim \mathcal{N}(0, 1)$, $a = \frac{2m-y-z}{\sqrt{2\theta}}$, $b = \frac{2M-y-z}{\sqrt{2\theta}}$ and where $\varphi(\cdot)$ is the probability density function of the standard normal law.

Proof 2

$$\begin{aligned}
\mathbb{E}\left[\frac{\partial r(X, y)}{\partial x} \frac{\partial r(X, z)}{\partial x}\right] &= \mathbb{E}\left[\frac{(X-y)(X-z)}{\theta^4} r(X, y)r(X, z)\right] \\
&= \mathbb{E}\left[\left(\frac{(X - \frac{y+z}{2})^2}{\theta^4} - \frac{1}{4} \frac{(y-z)^2}{\theta^4}\right) r(X, y)r(X, z)\right] \\
&= \mathbb{E}\left[\frac{(X - \frac{y+z}{2})^2}{\theta^4} r(X, y)r(X, z)\right] - \frac{1}{4} \frac{(y-z)^2}{\theta^4} \mathbb{E}[r(X, y)r(X, z)]
\end{aligned}$$

where

$$\mathbb{E}\left[\frac{(X - \frac{y+z}{2})^2}{\theta^4} r(X, y)r(X, z)\right] = \sqrt{\pi}\theta \exp\left(-\frac{(y-z)^2}{4\theta^2}\right) \underbrace{\mathbb{E}\left[\frac{(X - \frac{y+z}{2})^2}{\theta^4} \varphi_{\mathcal{N}(\frac{y+z}{2}, \frac{\theta^2}{2})}(X)\right]}_{\Delta}$$

with, using $w = (x - \frac{y+z}{2})/(\theta/\sqrt{2})$,

$$\begin{aligned}
\Delta &= (M-m)^{-1} \int_m^M \frac{(x - \frac{y+z}{2})^2}{\theta^4} \frac{1}{\sqrt{2\pi}\frac{\theta}{\sqrt{2}}} \exp\left(-\frac{(X - \frac{y+z}{2})^2}{2(\frac{\theta}{\sqrt{2}})^2}\right) dx \\
&= (M-m)^{-1} \int_a^b \frac{w^2}{2\theta^2} \varphi(w) dw \quad \text{with } a = \frac{2m-y-z}{\sqrt{2\theta}} \text{ and } b = \frac{2M-y-z}{\sqrt{2\theta}} \\
&= \frac{1}{2\theta^2(M-m)} \int_b^a w \varphi'(w) dw \\
&= \frac{1}{2\theta^2(M-m)} \left(a\varphi(a) - b\varphi(b) - \int_b^a \varphi(w) dw\right).
\end{aligned}$$