PHD THESIS DECLARATION

I undersigned

FAMILY NAME: *Lu*

NAME: *Xuefei*

Student ID no. *1824481*

Thesis title: *Sensitivity Analysis and Machine Learning for Computationally Challenging Computer Codes*

PhD in Statistics

Cycle 30^{th}

Student's Advisor: *Professor Emanuele Borgonovo*

Calendar year of thesis defence: *2019*

DECLARE

Under *my* responsibility:

1) that, according to Italian Republic Presidential Decree no. 445, 28th December 2000, mendacious declarations, falsifying records and the use of false records are punishable under the Italian penal code and related special laws. Should any of the above prove true, all benefits included in this declaration and those of the temporary "embarg" are automatically forfeited from the beginning;

2) that the University has the obligation, according to art. 6, par. 11, Ministerial Decree no. 224, 30th April 1999, to keep a copy of the thesis on deposit at the "Biblioteche Nazionali Central" (Italian National Libraries) in Rome and Florence, where consultation will be permitted, unless there is a temporary "embarg" protecting the rights of external bodies and the industrial/commercial exploitation of

the thesis;

3) that the Bocconi Library will file the thesis in its "Archivio Istituzionale ad Accesso Apert" (Institutional Registry) which permits online consultation of the complete text (except in cases of temporary "embarg");

4) that, in order to file the thesis at the Bocconi Library, the University requires that the thesis be submitted online by the student in unalterable format to Società NORMADEC (acting on behalf of the University), and that NORMADEC will indicate in each footnote the following information:

   - PhD thesis *Sensitivity Analysis and Machine Learning for Computationally Challenging Computer Codes*;
   - by *Lu, Xuefei*;
   - defended at Università Commerciale "Luigi Boccon"- Milano in the year *2019*;
   - the thesis is protected by the regulations governing copyright (Italian law no. 633, 22nd April 1941 and subsequent modifications). The exception is the right of Università Commerciale "Luigi Boccon" to reproduce the same, quoting the source, for research and teaching purposes;

5) that the copy of the thesis submitted online to Normadec is identical to the copies handed in/sent to the members of the Thesis Board and to any other paper or digital copy deposited at the University offices, and, as a consequence, the University is absolved from any responsibility regarding errors, inaccuracy or omissions in the contents of the thesis;

6) that the contents and organization of the thesis is an original work carried out by the undersigned and does not in any way compromise the rights of third parties (Italian law, no. 633, 22nd April 1941 and subsequent integrations and modifications), including those regarding security of personal details; therefore the University is in any case absolved from any responsibility whatsoever, civil, administrative or penal, and shall be exempt from any requests or claims from third parties;

7) that the thesis is not subject to "embarg", i.e. that it is not the result of work included in the regulations governing industrial property; it was not written as part of a project financed by public or private bodies with restrictions on the diffusion of the results; is not subject to patent or protection registrations.

22 January, 2019

Lu, Xuefei

Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my sincere gratitude and appreciation to all those who made this Ph.D. thesis possible.

First and foremost, I would like to gratefully acknowledge my advisor, Professor Emanuele Borgonovo, for introducing me to this exciting field of science, for his dedicated help, inspiration, encouragement and continuous support, for his continued patience reviewing and improving our papers and this thesis. He has provided me extensive personal and professional guidance and taught me a great deal about both scientific research and life in general. He has shown me, by his example, what a good scientist and person should be.

I am grateful for all faculty members of Decision Sciences Department at Bocconi University, who have helped and taught me immensely. I thank Professor Sonia Petrone, Director of the Ph.D. in Statistics, for the academic and financial support provided to carry out the research work. I would also like to acknowledge Professor Isadora Antoniano-Villalobos, who has always made herself available to clarify my doubts.

I would like to extend my gratitude to all of the researchers with whom I had the pleasure to work. I would especially like to thank Professor Mary C Hill, Professor Elmar Plischke, Professor Oldrich Rakovec, Professor Lorenzo Rosasco and Professor Alessandro Rudi.

My time at Bocconi was made enjoyable in large part due to my wonderful friends. I am grateful for the time spent with my colleagues Amir Khorrami Chokami, Stefano Rizzelli, Paolo Leonetti, Michele Peruzzi. I will always cherish the warmth shown by them. I also thank the Ph.D. students of other cycles who have been supportive in every way. A special mention of thanks to my friends, Zijian Wang and Veronica Cappelli for all the emotional support, entertainment, and caring they provided.

Last but not least, I would like to thank my parents for all their love and encouragement, for supporting me in all my pursuits. Many thanks also go to my loving and encouraging fiancé, Shanming Liu, for his faithful support during this Ph.D.

Abstract

This thesis contributes to sensitivity analysis for computer experiments. Computer experiments are becoming increasingly popular to support scientific investigations and decision-making. Thanks to recent advances in computing power, analysts are capable of building sophisticated computer codes that simulate the behavior of a system of interest. The simulator is considered as a black box and sensitivity analysis methods are essential to communicate insights about the input-output mapping to the analyst.

One of the main tasks of sensitivity analysis is to identify the key uncertainty drivers in the simulator response. However, this becomes a challenging task especially when the analyst is dealing with expensive-to-evaluate computer codes. Two main unresolved problems have been addressed in this thesis. First, the quantification of uncertainty in the estimates of global sensitivity measures at small sample sizes. Second, the creation of emulators for dimensionally large simulators. For the first task, a fully Bayesian approach to the estimation of global sensitivity measures is proposed. Four new classes of estimators are introduced, linking ideas in Bayesian non-parametrics to ideas in probabilistic sensitivity analysis. For the second task, an innovative Kriging emulator is proposed, borrowing recent advances from machine learning. The proposed emulator reduces the computational complexity in terms of time and memory requirements while achieving the same accuracy of currently implemented algorithms. Experiments show that the proposed algorithm offers significant improvements on simulators of increasing dimensionality.

Chapter 1

Introduction

Computer experiments are becoming increasingly important in scientific investigations. Computer models imitate the behavior of a physical or abstract system and allow the analysts to perform virtual experiments. In general, a computer model is used to map *assumptions* into *inferences*. The assumptions consist of the state-of-art understanding of the hypotheses, structures or inputs. The inferences refer to the produced outputs that are relevant to the analysis. Good modeling practices always select alternative assumptions and produce an interval of inferences, rather than mapping a single set of assumptions to a single inference.

Recent improvements in computing performance allow the analysts to create increasingly sophisticated computer models, usually associated with a dimensionality larger than in the past. Often, such complex simulators have no closed form solution and are considered as black boxes. The complexity of computer codes make it unfeasible to appreciate the input-output response solely based on intuition, which raises the need to look into the black box. The field of sensitivity analysis provides the analyst with a set of methods to gain insights into the model input-output relationship.

A broad range of applications have benefited from sensitivity analyses (Saltelli et al., 2006, Section 1). To name but a few, Saltelli and Tarantola (2002) apply sensitivity analysis to a nuclear risk assessment problem, Anderson et al. (2014) identify the most influential inputs for a climate change model, Hill et al. (2016) discuss the use of sensitivity analysis for model calibration of hydrological models. International agencies such as the US Environmental Protection Agency (US EPA, 2009), the British National Institute for Health Care Excellence (NICE, 2013) and the European Commission (2009), have suggested sensitivity analysis as an essential step in modeling process to guarantee the reliability and transparency of computer codes.

Sensitivity analysis serves as a link between the uncertainty about the assumptions and uncertainty about the inferences. As stated in Saltelli et al. (2006, p. 1113), “the scope of SA is not only to quantify and rank in order of importance the sources of prediction

uncertainty, but, which is much more relevant to calibration, to identify the elements (parameters, assumptions, structures, etc.) that are mostly responsible for the model realizations in the acceptable range. "In the literature, pioneers highlight the need to specify the objectives of sensitivity analysis (Saltelli et al., 2006, 2008; Borgonovo and Plischke, 2016). An objective, also called *setting*, is defined as "framing the sensitivity quest in such a way that the answer can be confidently entrusted to a well-identified measure "in Saltelli et al. (2008, p. 24). The objectives of sensitivity analysis include:

- *factor prioritisation*: to determine the most influential inputs, in the sense that, once determined, would lead to the greatest reduction in the output uncertainty;
- *factor fixing*: to fix the non-influential inputs for simulation simplification;
- *direction of change*: to appreciate if a change of the input would lead to an increase or decrease of the output;
- *interaction quantification*: to identify the interaction effects among inputs.

Over the years, plenty of sensitivity analysis techniques have been developed to address these sensitivity problems. Conventionally, such methods fall into two categories: local or global. Local sensitivity approaches focus on the simulation behavior around certain reference values of the inputs. Examples of local methods include Tornado diagrams for one-factor-at-a-time designs (Howard, 1988b), spider-plots (Eschenbach, 1992), derivative-based methods (Borgonovo and Apostolakis, 2001; Sobol' and Kucherenko, 2009; Rakovec et al., 2014). In between local and global methods, one can find Morris' screening method Morris (1991) and derivative-based global sensitivity measure (Kucherenko et al., 2009; Becker et al., 2018). Global sensitivity methods allow the inputs to vary in the entire input space. Typically, probability distributions are assigned to the inputs. Global methods include standardized regression coefficients, variance-based methods (Sobol', 1993), moment-independent methods (Borgonovo, 2007; Borgonovo et al., 2016), value of information (Strong et al., 2014) etc. In this thesis, global sensitivity analysis is the main focus, with special reference to the estimation of global sensitivity measures.

When a simulator is complex and expensive to evaluate, a single model evaluation may take minutes or even days, thus calling for the reduction in computational time. Meta-modeling or emulation tools are often used to reduce the computational burden. The intuition underlying the use of a meta-model (surrogate model or emulator) is to replace the original time-consuming computer code by a time-efficient emulator. Meta-modeling techniques such as Kriging (Gaussian process regression), polynomial chaos expansion (PCE), neural networks, and support vector machines are among the most popular practices (Sacks et al., 1989; Sudret, 2008; Santner et al., 2003). The applications of meta-modeling range from experimental design (Sacks et al., 1989; Welch et al.,

1992) to risk assessment and optimization problems (Kleijnen, 2017). In particular, it is used to reduce the computational burden for sensitivity analysis (Borgonovo et al., 2012).

1.1 Motivation and objectives

Quantifying uncertainty in a simulator predictor is essential. When there is only a limited number of model runs available due to the large dimensionality and long running times of the simulators, it is even more important to quantify uncertainty in the estimates of global sensitivity measures. To do this, one can resort to a Bayesian approach.

Computer experiments become complex and are characterized by a high number of inputs or realizations. However, conventional full-order emulators like kriging, are not suitable for such large datasets because they have been traditionally studied for problems in relatively small dimensions. Then, there is a need to study kriging emulators that can cope with the increased dimensionality. In this respect, recent advances have been made in the field of machine learning.

This thesis focuses on the following objectives:

1. to investigate and compare the application of alternative sensitivity analysis methods, focusing on the reduction of the computational burden;

2. to develop a fully Bayesian paradigm for sensitivity analysis using recent advances in Bayesian non-parametric techniques, such that the proposed methods allow uncertainty quantification in the estimates of global sensitivity measures for small sample sizes;

3. to use state-of-the-art machine learning techniques for meta-modeling, specifically for a popular surrogate model called Kriging, so that the resulting emulator is more efficient in terms of memory and time requirements;

4. to apply the proposed sensitivity analysis methods to complex realistic simulators.

1.2 Outline

The structure of this thesis is as follows.

Chapter 2 contains a thorough review of sensitivity analysis. As this chapter clearly shows, the proposals and methods in the sensitivity literature are plentiful. Thus, choosing among the competing methodologies requires a comprehensive understanding of their properties.

Chapter 3 concerns the global sensitivity analysis of complex hydrological simulators (Borgonovo et al., 2017). In particular, several sensitivity methods are investigated to obtain insights in response to four sensitivity settings simultaneously, without requiring any additional simulation run. To identify the key uncertainty drivers, we make use of variance-based and moment-independent sensitivity measures. To determine the trend, we make use of visualization tools such as the plots of the first order effects of the functional ANOVA expansion, the cumulative sum of normalized reordered output curves and scatter-plots of partial derivatives. To quantify interactions, we implement three emulators (PCE, high dimensional model representation, and LASI) to calculate higher-order global sensitivity measures.

Chapter 4 provides a Bayesian paradigm to conduct sensitivity analysis. In the literature, given-data (or one-sample) estimation is an approach that reduces computational burden notably. It is a post-processing technique that allows the analyst to obtain several sensitivity measures simultaneously from the same sample. The given-data method relies on an adequate partition of the input space. For a finite sample, the number of partition sets affects the estimates. In this work, we propose four Bayesian alternatives for estimating probabilistic sensitivity measures. Specifically, two Bayesian non-parametric bootstrap estimators are extensions of the one-sample estimators and are associated with two main features: 1) the non-parametric bootstrapping smooths the estimates within each partition set; 2) the estimators allow one to quantify the uncertainty in the estimates. However, these non-parametric bootstrap estimators do not avoid the issue of partition selection. Therefore, we introduce two estimators based on conditional and joint density estimation using Bayesian non-parametric approaches. These two estimators achieve the goal of providing credibility intervals of the quantity of interest without requiring a predefined partition.

Chapter 5 aims to improve computational efficiency and reduce memory requirements in Kriging meta-modeling. Kriging is commonly used for emulation in computer simulations. Kriging meta-modeling involves a Gaussian process with a correlation function evaluated at the data points, which becomes the so-called correlation matrix. The correlation matrix $\mathbf{R}$ plays a crucial role in Kriging calibration and has size $n \times n$, where n is the training sample size. The storage and inversion of $\mathbf{R}$ can be computationally infeasible for large n. The goal is to develop a new technique that deals with this issue. The idea is inspired on the Nyström regularization in machine learning. In Nyström regularization, instead of searching the optimal predictor in a space where kernels have size $n \times n$, the search is constrained to a space where kernels have size $n \times m$, where m represents the size of a randomly chosen subsample. It can be proved that the resulting new subroutine reduces computational cost from $O(n^3)$ to $O(nm^2)$, and memory requirements from $O(n^2)$ to $O(nm)$, without loss of accuracy. We have examined the performance of the proposed 'fast Kriging' on several complex and computationally intensive simulators used in op-

erations research, including the largest linear program in the Netlib library. The latter has $40,000$ inputs, which is a input dimension that is out of reach to this date. We also conduct the comparison of the proposed 'fast Kriging' with other subroutines currently in use. Results show a better performance of the new Kriging implementation with respect to existing ones.

Finally, Chapter 6 provides a brief discussion about potential directions for future research.

Chapter 2

Review

Sensitivity analysis methods have been studied widely over the past years. This chapter is dedicated to providing a review of the literature and unifying the framework for the various proposals.

Over the years, plenty of sensitivity analysis methods have been proposed, which possess diverse characteristics and are designed for different sensitivity settings. A systematic utilization of sensitivity analysis methods aims to "a) making model builders fully aware of the response of the model to variations and uncertainty in the model inputs and, b) providing decision makers with an enriched set of insights about the managerial problem at hand"(Borgonovo and Plischke, 2016, p. 884).

In general, sensitivity analysis methods can be classified into two categories: local and global. This classification is defined according to the region in which the inputs are allowed to vary. In this chapter, we first introduce several commonly used local methods; then describe some popular global sensitivity methods such as regression-based methods, the decomposition of the variance of model output; after which a common rationale of global sensitivity measures is presented. Finally, we discuss the use of meta-modeling techniques.

As the literature is vast, this chapter cannot claim exhaustiveness. For a more general overview, we refer to the review works of Borgonovo and Plischke (2016); Iooss and Lemaître (2015); Saltelli et al. (2006). Comprehensive contexts are offered by Saltelli et al. (2000, 2008); Borgonovo (2017).

2.1 Local sensitivity methods

In a local approach, the analyst is interested in studying the impact of inputs around specific locations (points of the model input space). The impact is identified through perturbations at the points of interest. In local sensitivity analysis, no probability distribution

is assigned to the inputs. For local sensitivity methods, we discuss the one-at-a-time design, scenario decomposition through finite changes, and screening methods.

2.1.1 One-at-a-time design

The most intuitive sensitivity approach is to assess the impact on the output when varying one input at a time. Assume one writes

$$\mathbf{y} = g(\mathbf{x}) + \epsilon(\mathbf{x}), \tag{2.1}$$

where g represents a set of operations performed by a computer code which processes a set $\mathbf{x}$ of inputs, resulting in a set $\mathbf{y}$ of outputs of interest; g is considered as a real-valued deterministic multivariate mapping, the input space is denoted as $\mathcal{X} \subseteq \mathbb{R}^k$ and output space as $\mathcal{Y} \subseteq \mathbb{R}^d$. The term $\epsilon(\mathbf{x})$ represents a zero-mean error term, which is present when the simulator response is stochastic. For simplicity, we focus on deterministic univariate responses, with $\epsilon(\mathbf{x}) \equiv 0$ and $d = 1$. Now consider evaluating the model at two locations (points) in $\mathcal{X}$, the base case $\mathbf{x}^0$ and a sensitivity (or alternative) case $\mathbf{x}^+$, where $\mathbf{x}^0 = (x_1^0, x_2^0, ..., x_k^0)$ and $\mathbf{x}^+ = (x_1^+, x_2^+, ..., x_k^+)$. Then, varying one input component at a time, one can quantify the individual effect of shifting x_i from x_i^0 to x_i^+ by the quantity $g(\mathbf{x}_i^+) - g(\mathbf{x}^0)$, where $\mathbf{x}_i^+ = (x_1^0, \ldots, x_{i-1}^0, x_i^+, x_{i+1}^0, \ldots, x_k^0)$. This is the so-called *one-at-a-time*, or *one-factor-at-a-time* design.

Tornado diagrams are the most successful graphical representation of one-at-a-time designs (Howard, 1988a; Eschenbach, 1992). To construct a Tornado diagram, one first defines three levels for each input, by assuming an additional sensitivity case denoted $\mathbf{x}^- = (x_1^-, x_2^-, ..., x_k^-)$. Then one can calculate the effects of two series of one-at-a-time sensitivities:

$$\Delta_i^- = g(\mathbf{x}_i^+) - g(\mathbf{x}^0) \quad \text{and} \quad \Delta_i^- = g(\mathbf{x}_i^-) - g(\mathbf{x}^0), \quad i = 1 \ldots k, \tag{2.2}$$

where $\mathbf{x}_i^- = (x_1^0, \ldots, x_{i-1}^0, x_i^-, x_{i+1}^0, \ldots, x_k^0)$. These quantities reflect the finite change in the output induced by the changes in x_i alone, and are called first order effects. The Tornado diagram is then constructed by arranging bar charts of Δ_i^- and Δ_i^+ separately in descending orders.For normalization purposes, one may divide Δ_i^+ by $|x_1^+ - x_i^0|/range(X_i)$, like in the method of Morris (Morris, 1991). The number of required simulation runs (computational cost) to construct a Tornado diagram is $C = 2k + 1$.

One-way sensitivity functions can be considered as a generalization of the one-at-a-time method. Instead of registering the impact of switching the input of interest x_i from the base value to a sensitivity value, an alternative is to consider varying this input over a predetermined range while the remaining simulator inputs are fixed at their base case

values. The one-way sensitivity function can be written as a function of x_i:

$$h(x_i) = g(x_1^0, \ldots, x_{i-1}^0, x_i, x_{i+1}^0, \ldots, x_k^0). \tag{2.3}$$

A graphically efficient visual representation of one-way sensitivity functions is offered by *spider plots* (Eschenbach, 1992), where the functions $h^*(x_i) = h(x_i) - g(\mathbf{x}^0)$ are displayed over a scaled range of inputs.

The methods mentioned above provide a rank of the inputs based on their impacts on a local scale. However, there are also limitations underlined in the literature, for example, the inability to detect interactions, or the inability to consider simultaneous changes of inputs, see Saltelli and Annoni (2010); Saltelli and D'Hombres (2010) for critiques of these basic methods.

2.1.2 Scenario decomposition

Scenario analysis has been widely studied and is becoming a popular decision tool in economics and strategic management (O' Brien, 2004; Tietje, 2005). In scenario analysis, decision-makers are interested in exploring the simulator responses on certain scenarios which are "descriptions of alternative hypothetical futures"(Jungermann and Thuring, 1988, p. 117). When dealing with quantitative simulators, the scenarios play the same role as the base/alternative cases. The decision-maker can consider a scenario as a point in the input space which draws particular attention. For a deterministic simulator, it has been proved that the change $\Delta g = g(\mathbf{x}^+) - g(\mathbf{x}^0)$ can be decomposed into $2^k - 1$ terms (Alis and Rabitz, 2001; Borgonovo, 2010). To illustrate, let us denote by $\mathbf{x}_{\mathbf{u}}^+$ a point where, $x_j = x_j^+$ if $j \in \mathbf{u}$, and $x_j = x_j^0$ otherwise, for $\mathbf{u} \subseteq \{1, 2, \ldots, k\}$ a subset of indices. Then the decomposition of Δg can be written as:

$$\Delta g = g(\mathbf{x}^+) - g(\mathbf{x}^0) = \sum_{i=1}^{k} \phi_i + \sum_{i<j} \phi_{i,j} + ... + \phi_{1,2,...k}, \tag{2.4}$$

where:

$$\begin{cases} \phi_i = g(\mathbf{x}_i^+) - g(\mathbf{x}^0), \\ \phi_{i,j} = g(\mathbf{x}_{i,j}^+) - \phi_i - \phi_j - g(\mathbf{x}^0), \\ \ldots \\ \phi_{\mathbf{u}} = g(\mathbf{x}_{\mathbf{u}}^+) - \sum_{i \in \mathbf{u}} \phi_i - \sum_{i,j \in \mathbf{u}, i<j} \phi_{i,j} - \cdots - g(\mathbf{x}^0) \\ \ldots \end{cases} \tag{2.5}$$

In the above system of equations, we arrange the elements in $\mathbf{u}$ in ascending order without replicates, i.e. for any $\phi_{\mathbf{u}=\{i,j,\ell\}}$, we have $i < j < \ell$. The terms $\phi_{\mathbf{u}}$ in (2.5) are called *finite change sensitivity indices* (Borgonovo, 2010). Note that ϕ_i coincides with Δ_i^+ of the

Tornado diagram, reflecting the individual effect of altering x_i from the base scenario to the alternative scenario. In case of multiple levels, the approach is still adaptable across each level. The higher-order terms $\phi_{i,j}, \phi_{i,j,\ell} \ldots$ measure the residual interactions provoked by the simultaneous variation of the corresponding inputs. Scenario decomposition merges the decomposition of the finite change and the idea of scenario analysis, introducing managerial interpretations to the analysis.

In principle, the calculation of finite change sensitivity indices of all orders requires $2^k - 1$ model evaluations. However, computational shortcuts are available in the literature (see Borgonovo (2010) for further details).

2.1.3 Screening methods

The goal of screening methods is to identify the least-influential inputs at a minimal cost with the purpose of discarding them for model simplification. One of the most successful screening methods is the method of Morris (Morris, 1991). The intuition is illustrated as follows. One considers l levels of each input, defining a grid in the input space. Then, the analyst randomly draws r points from the total l^k selected points, and performs a series of one-at-a-time designs. The individual effects are obtained by averaging over the r randomly drawn points. For each input, one can also calculate the standard deviation of the elementary effects obtained at the r points, which becomes a measure of non-linear and/or interaction effects. The computational cost of the Morris design is $C = r(k + 1)$. Additional technical details are found in Morris (1991); Campolongo and Saltelli (1997); Becker et al. (2018). The recently proposed enhanced version of elementary effects (Cuntz et al., 2015) allows for a more computationally efficient sequential screening.

Other screening methods in the literature include sequential bifurcation (Bettonvil, 1990; Bettonvil and Kleijnen, 1997), controlled sequential bifurcation (Wan, Ankenman, and Nelson, Wan et al.), the screening by groups (Dean and Lewis, 2006) etc. We refer to Campolongo et al. (2000); Kleijnen (2005); Woods and Lewis (2017) for thorough reviews.

2.2 Global sensitivity methods

While local methods involve the evaluation of the simulator at a limited number of locations, global sensitivity methods allow the inputs to vary over the entire space. The fundamental assumption of global sensitivity analysis is that we have knowledge of the input distributions, either joint or marginal, with or without correlations. The distribution information may come from the experts' knowledge or physical boundaries (Saltelli, 2002b). In this respect, another way to call this type of methods is *probabilistic sensitivity*

analysis (Oakley and O'Hagan, 2004), where inputs and output are considered as random variables, denoted as $\mathbf{X} = (X_1, \ldots, X_k)$ and Y.

In the *factor prioritization* setting, the analyst's goal is to identify the input parameter that, if fixed to a certain value, would lead to the greatest reduction in the variability of the model output (Saltelli and Tarantola, 2002). Thus, the degree of statistical dependence between Y and X_i is of concern. The stronger the statistical dependence, the more important we consider the parameter. Global sensitivity measures aim at summarizing such dependence accounting for uncertainty over the entire parameter support. In a variance-based sensitivity analysis (Ratto et al., 2007), the intuition is to quantify statistical dependence as the expected reduction in the simulator output variance due to fixing input X_i.

This section concisely discusses the most commonly used global sensitivity methods, including non-parametric methods, variance-based methods, and presents a common rationale for probabilistic sensitivity measures.

2.2.1 Non-parametric methods

The popularity of non-parametric techniques is due to the fact that these methods allow the analyst to derive sensitivity measures from a given sample. The term non-parametric is used in Saltelli and Marivoet (1990). In the literature, those methods are also called sampling-based methods. One of the earliest non-parametric measures is the well-known *Pearson correlation coefficient* (or Pearson product-moment correlation coefficient) of Y and X_i, developed by Karl Pearson in the 1880s. The definition of Pearson's correlation coefficient, usually denoted as ρ_{Y,X_i}, is

$$\rho_{Y,X_i} = \frac{\mathrm{Cov}(Y, X_i)}{\sigma_Y \sigma_{X_i}}, \tag{2.6}$$

where $\mathrm{Cov}(Y, X_i)$ denotes the covariance between Y and X_i, σ_Y and σ_{X_i} are the corresponding standard deviations. The absolute value of ρ_{Y,X_i} is less than or equal to one. In particular, a value of -1 or 1 implies a linear relationship between X_i and Y. Unfortunately, a null value of ρ_{Y,X_i} does not imply the independence between Y and X_i, e.g., $Y = X^2$.

Pearson correlation coefficient is closely related to regression-based indices. In particular when inputs are independent, ρ_{Y,X_i} coincides with the standardized regression coefficient (SRC) SRC_i. Regression-based methods rely on fitting a linear regression to a given sample. It is assumed that the input-output mapping can be well-represented by a

linear relationship $Y = b_0 + \sum_{i=1}^{k} b_i X_i$. In this case,

$$SRC_i = b_i \frac{\sigma_{X_i}}{\sigma_Y}, \quad i = 1 \dots k \tag{2.7}$$

are natural sensitivity measures because they account for the fractional reduction in the output variance. The reason is illustrated as follows. If the inputs are uncorrelated [1], the variance of Y can be decomposed as[2]:

$$\mathbb{V}[Y] = \sum_{i=1}^{k} b_i^2 \mathbb{V}[X_i]. \tag{2.8}$$

One can see that $b_i^2 \mathbb{V}[X_i]$ is the proportion of the variance of Y induced by X_i. The absolute value of SRC_i lies in $[0, 1]$, where a value close to 1 indicates that X_i has a major contribution to the variance of Y, and a value close to 0 implies little influence.[3]

However, if the linearity assumption is not valid, the conclusions derived by ρ_{Y,X_i} and SRC_i are not reliable. When the mapping is non-linear but still monotonic, one cannot directly use those regression-based indices but may resort to a pre-processing the dataset by rank transformations, that is, replacing the values of the input-output realizations by their ranks in the corresponding dimensions (Saltelli et al., 2000). The resulting sensitivity measures become the *Spearman's rank correlation coefficients* and *standardized rank regression coefficients*. Note that the monotonic hypothesis in the rank-transformed data needs to be validated.

When no hypothesis can be made regarding the model structure, a more general method is needed. Researchers have proposed the *functional ANOVA decomposition*, which we discuss in the next section.

2.2.2 Functional ANOVA decomposition and variance-based sensitivity indices

To illustrate, we denote the input probability space by $(\mathcal{X}, \mathcal{B}(\mathcal{X}), \mathbb{P}_{\mathbf{X}})$, where $\mathbb{P}_{\mathbf{X}}$ represents the joint probability measure of $\mathbf{X} = (X_1, \dots, X_k)$, assumed known. Similarly, $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}), \mathbb{P}_Y)$ denotes the output probability space, where $\mathbb{P}_Y$ represents the distribution of Y induced by $\mathbb{P}_{\mathbf{X}}$. Let us consider $g(\mathbf{X})$ is a real-valued measurable function on $\mathcal{X}$ and

[1] In general, SRC is defined under classical assumptions for regression analysis, where X_i are assumed to be linearly independent.

[2] Both $\mathbb{V}[Y]$ and σ_Y^2 denote the variance of Y.

[3] Note that the square of the correlation coefficient is R^2.

is square integrable

$$g \in \mathcal{L}^2(\mathcal{X}, \mathcal{B}(\mathcal{X}), \mathbb{P}_{\mathbf{X}}) = \left\{ g : \int_{\mathcal{X}} g(\mathbf{x})^2 \mathrm{d}\mathbb{P}_{\mathbf{X}} < \infty \right\}.$$

Under the assumption of a product probability measure $\mathrm{d}\mathbb{P}_{\mathbf{X}}(\mathbf{x}) = \prod \mathrm{d}\mathbb{P}_{X_i}(x_i)$, and the *strong vanishing conditions*:

$$\int_{\mathcal{X}} g_{\mathbf{u}}(\mathbf{x}_{\mathbf{u}}) \mathrm{d}\mathbb{P}_{X_i}(x_i) = 0, \quad i \in \mathbf{u} \neq \emptyset, \quad \mathbf{u} \subseteq \{1, 2, \ldots, k\},$$

g can be decomposed exactly into 2^k components (Efron and Stein, 1981):

$$\begin{aligned} g(\mathbf{X}) &= \sum_{\mathbf{u} \subseteq \{1,\ldots,k\}} g_{\mathbf{u}}(\mathbf{X}_{\mathbf{u}}) \\ &= g_0 + \sum_{i=1}^{k} g_i(X_i) + \sum_{1 \leq i < j \leq k} g_{i,j}(X_i, X_j) + \cdots + g_{1,\ldots,n}(X_1, \ldots, X_k) \end{aligned} \tag{2.9}$$

where

$$\begin{aligned} g_0 &= \mathbb{E}_{\mathbf{X}}\left[g(\mathbf{X})\right] = \int g(\mathbf{x}) \mathrm{d}\mathbb{P}_{\mathbf{X}} \\ g_i(X_i) &= \mathbb{E}_{\mathbf{X}_{-i}}\left[g(\mathbf{X})|X_i\right] - g_0 = \int g(\mathbf{x}_{-i}, X_i) \mathrm{d}\mathbb{P}_{X_{-i}}(\mathbf{x}_{-i}) - g_0 \\ g_{i,j}(X_i, X_j) &= \mathbb{E}_{\mathbf{X}_{-\{i,j\}}}\left[g(\mathbf{X})|X_i, X_j\right] - g_i(X_i) - g_j(X_j) - g_0 \\ &= \int g(\mathbf{x}_{-\{i,j\}}, X_i, X_j) \mathrm{d}\mathbb{P}_{X_{-\{i,j\}}}(\mathbf{x}_{-\{i,j\}}) - g_i(X_i) - g_j(X_j) - g_0 \\ &\cdots \\ g_{1,2,\ldots,k}(\mathbf{X}) &= g(\mathbf{X}) - \sum_{j=1}^{k-1} \sum_{i_1 < i_2 < \cdots < i_j} g_{i_1,\ldots,i_j}(X_{i_1}, X_{i_2}, \ldots, X_{i_j}) - g_0. \end{aligned} \tag{2.10}$$

In eq (2.10), $\mathbf{X}_{-i}$ is a shorthand for $(X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_k)$, and $\mathrm{d}\mathbb{P}_{X_{-i}}$ is short for $\prod_{j \neq i} \mathrm{d}\mathbb{P}_{X_j}$; g_0 is the expectation of Y; $g_i(X_i)$ is called the *first order effect function* and displays the expected behavior of Y as a function of X_i; $g_{i,j}(X_i, X_j)$ is the interaction effect between X_i, X_j, etc. The generic effect function $g_{\mathbf{u}}(\mathbf{x}_{\mathbf{u}})$ has null expectation and two generic effect functions are mutually orthogonal (Sobol', 1993; Rabitz and Alis, 1999). Note that, if we consider the above decomposition at two specific points $\mathbf{x}^0$ and $\mathbf{x}^+$, one obtains a 2^k term decomposition of a finite change under $\mathbb{P}_{\mathbf{X}}$.

The orthogonality and null expectation properties of the terms in Eq. (2.10) allow to obtain the complete decomposition of the variance of Y as (Efron and Stein, 1981; Sobol',

1993):

$$\mathbb{V}[Y] = \sum_{i=1}^{k} V_i + \sum_{i<j}^{k} V_{i,j} + \cdots + V_{1,2,\ldots,k}, \tag{2.11}$$

where

$$\begin{aligned} V_i &= \int g_i^2 \mathrm{d}\mathbb{P}_{X_i} \\ V_{i,j} &= \iint g_{i,j}^2 \mathrm{d}\mathbb{P}_{X_i} \mathrm{d}\mathbb{P}_{X_j} \\ &\cdots \\ V_{1,2,\ldots,k} &= \int \cdots \int g_{1,2,\ldots,k}^2 \mathrm{d}\mathbb{P}_{\mathbf{X}}. \end{aligned} \tag{2.12}$$

In Eq. (2.12), the first order terms V_i account for the portion of the total model output variance $\mathbb{V}[Y]$ due to the individual variation of X_i; the second order terms $V_{i,j}$ represent the part apportioned by the residual interaction of inputs X_i and X_j. A similar interpretation applies to higher order terms. The variance based sensitivity indices are defined as(Sobol', 1993; Homma and Saltelli, 1996):

$$S_{\mathbf{u}} = \frac{V_{\mathbf{u}}}{\mathbb{V}[Y]}. \tag{2.13}$$

The quantity $S_{\mathbf{u}}$ is called the variance-based sensitivity index of group $\mathbf{u} \subseteq \{1, 2, \ldots, k\}$ and it represents the fractional contribution to the model output variance resulting from the interaction between inputs $X_{\mathbf{u}}$. When the group $X_{\mathbf{u}}$ is restricted to an individual input X_i, we obtain the first-order Sobol' index defined as

$$S_i = \frac{V_i}{\mathbb{V}[Y]}. \tag{2.14}$$

The index S_i corresponds to the fraction of $\mathbb{V}[Y]$ associated with the individual contribution of X_i. If one wishes to consider the overall contribution of X_i, one needs to account also for its interactions with the remaining inputs. One writes:

$$ST_i = \frac{V_i + \sum_{i \neq j}^{k} V_{i,j} + \cdots + V_{1,2,\ldots,k}}{\mathbb{V}[Y]}. \tag{2.15}$$

The index ST_i is, then, the total fractional contribution of X_i to the variance of Y. For independent inputs, we have $\sum S_{\mathbf{u}} = 1$ and $\sum_{i=1}^{k} ST_i \geq 1$. In particular, we have that $ST_i = S_i$ if X_i is not involved in interactions with other inputs. If this occurs for all inputs, then we have that $\sum_{i=1}^{k} S_i = 1$, so that $S_i^{\text{Interaction}} = 0$ and the model response is additive. Finally, for computational assessment of different numerical approaches to calculate total (variance-based) interaction indices we refer to Fruth et al. (2014).

The ANOVA decomposition for correlated inputs is studied in (Rahman, 2014; Li and Rabitz, 2012).

2.2.3 Global sensitivity measures: a common rationale

In a variance-based sensitivity analysis (Ratto et al., 2007), the intuition is to quantify statistical dependence as the expected reduction in model output variance due to fixing input X_i. Following this intuition, we can define the expected reduction in simulator output variance as (Homma and Saltelli, 1996):

$$\eta_i = \mathbb{E}\left[\frac{\mathbb{V}[Y] - \mathbb{V}[Y|X_i]}{\mathbb{V}[Y]}\right] = \frac{\mathbb{V}[Y] - \mathbb{E}[\mathbb{V}[Y|X_i]]}{\mathbb{V}[Y]} = \frac{\mathbb{V}[\mathbb{E}[Y|X_i]]}{\mathbb{V}[Y]}, \tag{2.16}$$

where $\mathbb{V}[Y|X_i]$ represents the variance of Y given that X_i is fixed. The definition of η_i in Eq. (2.16) is equivalent to the first-order Sobol' indices S_i in Eq. (2.14) when inputs are mutually independent. Borgonovo et al. (2016) point out that the rationale at the basis of variance-based sensitivity measures is common to other sensitivity measures. Eq. (2.16) can be seen as a measure of the separation between the marginal model output distribution ($\mathbb{P}_Y$) and the conditional model ($\mathbb{P}_{Y|X_i=x_i}$) output distribution given X_i in terms of variance reduction. In general, we can consider $\zeta(\mathbb{P}_Y, \mathbb{P}_{Y|X_i=x_i})$ a (discrepancy) operator between probability measures over $\mathcal{Y}$, which is evaluated at the marginal-conditional pair $\mathbb{P}_Y$ and $\mathbb{P}_{Y|X_i=x_i}$. The expectation of a measure of discrepancy:

$$\xi_i := \mathbb{E}[\zeta(\mathbb{P}_Y, \mathbb{P}_{Y|X_i})], \tag{2.17}$$

where the expectation is calculated with respect to the marginal distribution of X_i, is called the *probabilistic sensitivity measure* of X_i with *inner operator* ζ (Borgonovo et al., 2014). In order to be a sensible measure of the discrepancy between $\mathbb{P}_Y$ and $\mathbb{P}_{Y|X_i=x_i}$, the operator $\zeta(\cdot,\cdot)$ is required to be null when the two distributions are identical, i.e. $\zeta(\mathbb{P},\mathbb{P}) = 0$.

We denote the joint cumulative distribution function of the inputs by $F_{\mathbf{X}}$, if the density exists, we denote the joint density as $f_{\mathbf{X}}$, and the marginal cdf and pdf of X_i as F_{X_i} and f_{X_i}.The cdf and pdf of the model output are denoted by F_Y and f_Y, respectively. Examples of inner operators used in the literature are:

$$\zeta^V(\mathbb{P}_Y, \mathbb{P}_{Y|X_i}) = \frac{\mathbb{V}[Y] - \mathbb{V}[Y|X_i]}{\mathbb{V}[Y]}, \tag{2.18}$$

for variance-based sensitivity measures (Homma and Saltelli, 1996), the L^1-distance be-

tween density functions,

$$\zeta^{L1}(\mathbb{P}_Y, \mathbb{P}_{Y|X_i}) = \int_{\mathcal{Y}} \mid f_Y(y) - f_{Y|X_i}(y|X_i) \mid \mathrm{d}y, \tag{2.19}$$

for the δ-importance measure (Borgonovo, 2007), the Kolmogorov-Smirnov (KS) distance between cdfs in the PAWN method (Pianosi and Wagener, 2015),

$$\zeta^{KS}(\mathbb{P}_Y, \mathbb{P}_{Y|X_i}) = \sup_{\mathcal{Y}} \mid F_Y(y) - F_{Y|X_i}(y|X_i) \mid, \tag{2.20}$$

or the Kuiper distance in Baucells and Borgonovo (2013),

$$\zeta^{KU}(\mathbb{P}_Y, \mathbb{P}_{Y|X_i}) = \sup_{\mathcal{Y}} \left(F_Y - F_{Y|X_i}\right) + \sup_{\mathcal{Y}} \left(F_{Y|X_i} - F_Y\right). \tag{2.21}$$

Baucells and Borgonovo (2013) and Borgonovo et al. (2014) discuss the use of distances between cumulative distribution functions in sensitivity analysis proposing a general definition of which Eqs. (2.20) and (2.21) are particular cases. Kuiper (1960) underlines that some limitations associated with the KS distance can be overcome by its modification in the Kuiper metric. In particular, the Kuiper distance "puts all percentiles on equal footing "(Crnkovic and Drachman, 1996, p. 140). Alternative choices of ζ lead to different global sensitivity measures. For instance, in the PAWN method (Pianosi and Wagener, 2015) the inner operator is the Kolmogorov-Smirnov distance between cumulative distribution functions and the statistic is the median. The variance-based sensitivity measure is obtained by taking the expectation of Eq. (2.18); similarly, δ_i, β_i^{KS} and β_i^{KU} can be obtained by taking the expectation of Eqs. (2.19), (2.20) and (2.21) respectively (Borgonovo et al., 2016):

$$\delta_i = \frac{1}{2}\mathbb{E}_{X_i}\left[\int_{\mathcal{Y}} \mid f_Y(y) - f_{Y|X_i}(y|X_i) \mid dy\right], \tag{2.22}$$

$$\beta_i^{KS} = \mathbb{E}_{X_i}\left[\sup_{\mathcal{Y}} \left|F_Y(y) - F_{Y|X_i}(y)\right|\right], \tag{2.23}$$

$$\beta_i^{KU} = \mathbb{E}_{X_i}\left[\sup_{\mathcal{Y}} \left(F_Y - F_{Y|X_i}\right) + \sup_{\mathcal{Y}} \left(F_{Y|X_i} - F_Y\right)\right]. \tag{2.24}$$

2.2.4 Properties of global sensitivity measures

In this section, we limit ourselves to two important properties: nullity-implies-independence and monotonic transformation invariance.

The *nullity-implies-independence property*, is also known as Rényi's postulate D for measures of statistical dependence (Renyi, 1959). This property is relevant in measuring statistical dependence and is defined in such a way that a null value of a global sensitivity

measure reassures the analyst that Y is independent of X_i. This is not necessarily true in general. For instance, although variance-based sensitivity measures are widely-used in the literature, they do not possess this property. Plischke et al. (2013, proposition 1) consider functions of the form

$$y = g(\mathbf{x}) = a(\mathbf{x}_{\mathbf{u}})h(x_j) + b(\mathbf{x}_{\mathbf{u}'}), \text{with} \quad \mathbf{u} \cup \{j\} \cup \mathbf{u}^{'} = \{1 \dots k\}, \tag{2.25}$$

where $j \notin \mathbf{u} \cup \mathbf{u}^{'}$ and $\mathbf{u} \cap \mathbf{u}^{'} = \emptyset$. Then if $\mathbb{E}[h(X_j)] = 0$, the first-order indices η_i of Eq. (2.25) are all zeros for any $i \in \mathbf{u}$. That is, for a function of the form in Eq. (2.25), even if Y is statistically dependent on $X_{\mathbf{u}}$, such dependence is not revealed by η_i (Note that the total indices ST_i account for interactions). A widely studied example named *Ishigami function* (Ishigami and Homma, 1990; Saltelli et al., 2004) falls in this family of functions

$$Y = \sin X_1 + 7 \sin^2 X_2 + 0.1(X_3)^4 \sin X_1 = \left(1 + 0.1(X_3)^4\right) \sin X_1 + 7 \sin^2 X_2 \tag{2.26}$$

with $\mathbf{u} = \{3\}$, $j = 1$.

A second relevant property is monotonic transformation invariance. This property is particularly useful for estimation, because transformation of the output (typically a logarithmic or a rank transformation) is shown to accelerate numerical convergence in some cases(Borgonovo et al., 2014). To illustrate, let us denote by $\xi(Y)$ a generic sensitivity measures calculated on output Y, by $\xi(T)$ the same sensitivity measure calculated on the transformed output $T = t(Y)$, where $t(\cdot)$ is a monotonically increasing (decreasing) function, e.g. $T = \log(Y)$. We say that a global sensitivity measure ξ_i is monotonic transformation invariant, if $\xi_i(Y)$ is equal to $\xi_i(T)$. Clearly, issues emerge if one transforms the output but then uses a sensitivity measure which is not transformation invariant, since the results obtained on the transformed scale might not be directly transferred back to the original scale. This problem is, of course, avoided if one uses a transformation invariant sensitivity measure. Borgonovo et al. (2014) offer some analytical examples that demonstrate the issues caused by the use of transformations. For example, they consider a model

$$Y = \exp(X_1 + 2X_2) \quad \text{with} \quad X_1, X_2 \overset{iid}{\sim} \mathcal{N}(0, 1).$$

The above function contains interactions, with a second order interaction effect $S_{1,2}(Y) = 0.625$. After a logarithmic transformation on Y, $S_{1,2}(T)$ becomes zero, because the transformed model becomes additive. Thus, information about interactions in the model structure is different before and after the transformation.
Furthermore, transformations can also cause rank reversals. If one considers the following

function (Borgonovo et al., 2014)

$$Y = X_1(3 + X_2^{X_3}) \quad \text{with} \quad X_1 \sim U(1, 100), X_2 \sim U(400, 600), X_3 \sim U(0.5, 1),$$

where $U(a, b)$ stands for a uniform distribution over interval $[a, b]$, one has $S_3(Y) = 0.54 > S_1(Y) = 0.26$. This result indicates that X_3 is significantly more influential than X_1. However, after a logarithmic transformation, one obtains $S_3(T) = 0.486 < S_1(T) = 0.499$, so that X_3 appears less influential than X_1 after the transformation. The problem of rank reversal is avoided if one uses a transformation invariant sensitivity measure.

The properties of a global sensitivity measure are strictly related to the choice of the inner operator $\zeta(\cdot, \cdot)$ in Eq. (2.17). Generally, if one chooses an inner operator based on the family of Csiszár divergences, or *f-divergences* (Rahman, 2016), one is reassured to obtain a sensitivity measure that possesses both the nullity-implies-independence and the transformation invariance properties. All global sensitivity measures mentioned so far, with the exception of variance-based sensitivity measures, i.e. $\delta_i, \beta_i^{KS}, \beta_i^{KU}$, possess the nullity-implies-independence and monotonic transformation invariant properties. We refer to Borgonovo et al. (2014) and Borgonovo et al. (2016) for further details.

2.2.5 Estimation of global sensitivity measures

The estimation of global sensitivity measures is a challenging task. The most intuitive way is to adopt a *brute force* approach, where estimators are defined strictly following the definition in Eq. (2.17). The brute force method involves a Monte Carlo estimation of the outer expectation with n_{ext} simulation runs in Eq. (2.17), and additional n_{int} runs for calculation of the inner statistics. Specifically, for each i, n_{ext} samples of X_i from its marginal and for each of those, n_{int} simulations of X_{-i} from $P_{X_{-i}}$ leading to the total number of model evaluations $C^{BF} = k \cdot n_{ext} \cdot n_{int}$.

The advantage of the brute force approach is that, under unbiased estimation of the required distributions, it leads to unbiased estimators. However, when inputs are not independent, sampling from the conditional distributions $f_{\mathbf{X}_{-i}|X_i=x_i}$ may not be feasible. In fact, $f_{\mathbf{X}_{-i}|X_i=x_i}$ may not be available. Even if we have a closed form for $f_{\mathbf{X}_{-i}|X_i=x_i}$, sampling from it can also be challenging due to its complexity. Techniques like importance sampling or Gibbs sampling can be adopted, however, the implementation of a brute force approach in the presence of dependence might be cumbersome. Another disadvantage of a brute force approach is the associated computational cost. Because several global sensitivity measures require the estimation of a density or of a cdf, n_{ext} and n_{int} need to be sufficiently large. Previous studies consider n_{ext} and n_{int} of the order of hundreds or thousands of simulation evaluations, with an overall computational cost of hundreds of

thousands simulator runs. For computationally intensive simulators, the calculation of global sensitivity measures then becomes infeasible. Resorting to an emulator [4] becomes the only way by which an analyst can compute global sensitivity measures (see Ratto and Pagano (2010); Borgonovo et al. (2012) among others).

In the literature, researchers have devoted efforts to the creation of designs that may abate such computational costs. Investigators have tackled the estimation of variance-based sensitivity measures since the late 1990's. For instance, Saltelli et al. (1999); Saltelli (2002b) introduce the extended Fourier amplitude sensitivity test (FAST) method which enables the estimation of variance-based sensitivity measures at a cost of $C = kn$ simulation runs. Recently, the given-data (or one-sample) approach has been a subject of investigation, because it has the potential of abating the computational cost to n simulation runs. A given-data design is not specific to a particular sensitivity measure, but is suitable for any sensitivity measure in the form of Eq. (2.17) (Borgonovo et al., 2016). The principle can be dated back to Pearson (1905) and is illustrated as follows. Given a size n dataset of input-output realizations denote by $Data = \{(\mathbf{x}^j, y^j) : j = 1, \dots n\}$, where $\mathbf{x}^j = (x_1^j, \dots, x_k^j)$; the subscript stands for the i-th input, and the superscript for the j-th realization, one first partitions the support of X_i into M bins $\{\mathcal{X}_i^m, m = 1 \dots M\}$; The key-intuition is then to use the bin conditions $\widehat{\mathbb{P}}_{\mathbf{Y}|X_i \in \mathcal{X}_i^m}$ instead of the point conditions $\widehat{\mathbb{P}}_{\mathbf{Y}|X_i=x_i}$ in the estimator.

The given-data approach provides two main advantages. First, it allows to post-process the sample so that several sensitivity measures can then be estimated simultaneously. Second, nominally, the given-data approach reduces the computational cost to n model evaluations, and the computational cost is independent of the number of inputs. In particular, one may want to discuss the efficiency under a limited budget of model runs. Suppose that the minimum budget of other estimators except for the given-data is $k \times n^{others}(k)$, where n depends on the number of input dimensions k. To make given-data estimator less efficient than other estimators, one should have that, for reaching the same level of accuracy, $n^{given-data}(k) >> k \times n^{others}(k)$. This implies that we have $n^{others} << n^{given-data}/k$. Suppose k is of the order of 100 (in this thesis, simulators with k=40,000 were addressed), then, a method whose cost depends on k should be capable of reaching the same accuracy as a given data approach with $n^{other} = n^{given-data}/100$. Intuitively, this seems a difficult condition to be reached, especially if the output of the simulator is sparse. For example, if the budget is $n = 10000$, applying a given-data estimator one would use all model runs for each input. If now one uses a double-loop estimator, assuming there are 10 inputs, then one has only 10 points for each inner loop, which is hardly more 'efficient' than the given-data estimator, unless the experiment design is super perfect for the problem. Therefore, nominally, the computational cost of given-data

[4] Emulator is also called metamodel, or surrogate model in this work, see Section 2.3 for a brief discussion.

estimation is independent to the number of inputs. Given-data (or one-sample) estimators have been successfully applied for the computation of variance-based (Strong et al., 2012), value of information (Strong and Oakley, 2013) and other sensitivity measures, such as $\delta_i, \beta_i^{KS}, \beta_i^{KU}$ (Plischke et al., 2013; Borgonovo et al., 2014). In Plischke et al. (2013) the analysis is conducted for an 800 input model, while in Plischke and Borgonovo (2017) for a $30,000$ input simulator.

The estimation of global sensitivity measures is one of main focuses of this thesis, therefore, a separate section in Chapter 4 is devoted to the discussion in greater details.

2.3 Meta-models

Meta-modeling techniques have been developed across disciplines, ranging from statistics and mathematics, to engineering and machine learning. Analysts use meta-models as surrogates of the time-consuming computer code to reduce the overall computational burden.

In the literature, meta-modeling plays a role in various research areas. For example, in design space exploration, the analyst works with a cheap-to-run meta-model to explore the design space in order to enhance the understanding of the design problem (Simpson et al., 2001; Kleijnen, 2017). In an optimization problem, the analyst may reduce the search range by the assistance of a meta-model. In sensitivity analysis, the analyst can extract the input-output insights through a computationally cheap surrogate model instead of directly using the expensive simulator (Borgonovo et al., 2012).

Meta-model approximation or meta-modeling is the key to emulation-based applications. In this thesis, meta-modeling refers to the mathematical discipline that has an interest in emulating the statistical input-output mapping from a set of observations and a limited set of prior assumptions. In machine learning, it is often known as *supervised learning* (Williams and Rasmussen, 2006) [5]. Supervised learning is generally divided into two categories based on the nature of the output, namely, regression and classification. In regression problems, the output is continuous, while in classification problems, the output takes value in a discrete set of labels. In this thesis, we focus on regression problems.

There exists a wide variety of emulators for regression problems. One of the best-known methods is generalized linear regression. Numerous studies have contributed to the use of linear regression to construct emulators, and we refer to the monographs of Kutner et al. (1996) and Chatterjee and Hadi (2006). Neural networks are also popular emulators, see Rojas (1996); LeCun et al. (2015); Lampinen and Vehtari (2001); Ripley

[5] Unsupervised learning refers to the discipline that has interest in building statistical models for inputs, but without any supervision of outputs/targets.

(1994). Polynomial chaos expansion is another popular method and we refer the readers to Blatman and Sudret (2010); Sudret (2008); Schobi et al. (2015) for reviews. Other types of emulators include smoothing spline ANOVA meta-models (Ratto et al., 2007), radial basis functions (RBF) (Dyn et al., 1986; Fang and Horstemeyer, 2006), multivariate adaptive regression splines (MARS) (Friedman, 1991), support vector regressions (SVR) (Clarke et al., 2005) etc. Sacks et al. (1989) introduce the use of Kriging meta-modeling. In Kriging, the deterministic output response is considered as a realization of a Gaussian process. We refer to Chapter 5 of this thesis, which is devoted to the development of a fast Kriging method, for further details.

Researchers have developed various types of fitting methods in the literature. Each meta-model often has its associated fitting method. For instance, the (weighted) least square method is frequently used for generalized linear regression; the best linear unbiased predictor (BLUP) is usually preferred for Kriging; back propagation is used for artificial neural networks. We refer to Simpson et al. (2001) and Wang (2007) for a detailed review of common fitting methods.

Meta-model validation is an essential step before the fitted surrogate model can be used. Two classes of commonly used methods are in-sample cross-validation and out-of-sample validation. The p-fold cross-validation belongs to the first class, the idea is as follows: one first divides the training set into p subsets, then fits the meta-model p times, each time, leaving one of the subsets out of training and using the omitted subset to compute the fitting error Meckesheimer et al. (2002). The leave-q-out method is a variation of p-fold cross-validation, where all possible subsets of size q are considered. The out-of-sample validation employs additional points to measure the meta-model accuracy, e.g. the root mean square error (RMSE), the maximum absolute error (MAX) and the coefficient of determination (also known as R^2). RMSE is used to measure the overall accuracy of the meta-model, while MAX focuses on the local accuracy.

One can work with meta-models by adopting different strategies. The most commonly adopted strategy is to first fit a meta-model, then use the fitted emulator to replace the computationally expensive simulator (Borgonovo et al., 2012). Another popular strategy involves an iterative process with validation or optimization steps in the loop to decide the re-sampling or re-fitting. At each iteration, a new sample is generated, and the meta-model is updated to improve or maintain the model accuracy (Jones et al., 1998). These sampling approaches are often called 'batch', sequential/ adaptive sampling approaches.

The literature on emulators is broad and cannot be exhaustively developed in this section. We refer to Wang and Shi (2013); Wang (2007) for meta-modeling techniques in optimization problems, to Kleijnen (2017) for a review of meta-modeling from a design of experiments perspective, to Simpson et al. (2001) for an empirical comparison of the performance of alternative meta-models, and to the monographs of Santner et al. (2003); Hastie et al. (2009) and Williams and Rasmussen (2006).

Chapter 3

Sensitivity analysis of complex hydrological simulators

This chapter investigates sensitivity analysis methods for gaining greater insight from hydrological simulation runs conducted for uncertainty quantification and model differentiation. We frame the sensitivity analysis questions in terms of the main purposes of sensitivity analysis: parameter (factor) prioritisation, trend identification and interaction quantification. For parameter prioritisation, we consider variance-based sensitivity measures, sensitivity indices based on the L^1-distance, the Kuiper metric and the sensitivity indices of the DELSA methods. For trend identification, we investigate insights derived from graphing the one-way ANOVA sensitivity functions, the recently introduced CUSUNORO plots and derivative scatter plots. For interaction quantification, we consider information delivered by variance-based sensitivity indices. In this work, we apply the so-called given-data principle, which allows one to perform the above analyses from a set of simulation runs. Therefore, one avoids using specific designs for each insight, thus controlling the computational burden. The methodology is applied to a hydrological simulator of a river in Belgium simulated using the well established Framework for Understanding Structural Errors (FUSE) on five alternative configurations. The findings show that the integration of the chosen methods provides insights unavailable in most other analyses.

This chapter contains joint work with Emanuele Borgonovo, Mary C. Hill, Elmar Plischke, and Oldrich Rakovec, and is based on Borgonovo et al. (2017).

3.1 Motivation

This study focuses on sensitivity analysis of hydrological models [1], which are used to analyze water shortage (drought), water excess (flood), water quality (contamination of drinking water and/or crops), and river dynamics (erosion). These can cause large socio-economic damage and ways to prevent such damage are of intense interest. Computer models are developed with the hope of adequately representing the real world complexity of rainfall-runoff processes in hydrology catchments, the contributing areas from which a given stream/river derives its flow. In this context, Gupta et al. (2012); Foglia et al. (2013) adequately suggest a level of accuracy that makes model results useful for managing the system being simulated. Model inputs/parameters generally cannot be directly measured in nature with sufficient accuracy, and therefore are commonly estimated through inverse modeling — see, among others Duan et al. (1992); La Vigna et al. (2016). The process has inherent uncertainty, and measures of uncertainty commonly accompany any modeling analysis (e.g., Pappenberger and Beven, 2006; Montanari, 2007; Beven, 2011; Nearing et al., 2016). Sensitivity analysis is conducted to understand the relation between inputs and outputs and to obtain insights in what often is a complicated model input-output mapping (Hill and Tiedeman, 2007; Saltelli et al., 2008; Rosero et al., 2010; Mendoza et al., 2015; Norton, 2015; Hill et al., 2016; Pianosi et al., 2016; Razavi and Gupta, 2016a; Markstrom et al., 2016; Houle et al., 2017). These discoveries help the analyst to use simulated values appropriately in planning, risk assessment, and decision support.

In the simulation of environmental/hydrologic systems, the plethora of sensitivity analysis methods that have been developed causes confusion and for many of the methods execution times can be colossal (e.g., Hill et al., 2016, and references cited therein). All this recommends careful consideration of the necessary model runs and how they are used. Of interest are studies that explore how a set of model runs can be used to obtain relevant and varied insights. In this work we consider the utility of a set of sensitivity analysis methods.

Our approach rests on two main pillars: clearly stating the sensitivity analysis goals from the start and controlling the computational burden.Regarding goals, we make use of the methodology of sensitivity analysis settings. “A setting is used to frame the sensitivity quest in such a way that the answer can be confidently entrusted to a well-identified measure”(Saltelli et al., 2008, p. 24). We consider the following three sensitivity settings which have emerged from previous sensitivity analysis studies: *parameter prioritisation* (Ratto et al., 2007), *trend identification* and *interaction quantification* — see, among others Borgonovo and Plischke (2016)). These settings can be used at different stages of the modeling process. Saltelli et al. (2000) and Hill and Tiedeman (2007) emphasize the role

[1] In this chapter, following the conventions in hydrological modeling, the word ‘model’ refers to the simulator/computer code, and the word ‘parameter’ refers to an input or input factor.

of sensitivity analysis throughout model development, starting with the model building phase. Ratto et al. (2007) discuss the role of sensitivity analysis to support calibration and validation. As an example, consider using the root mean square error (RMSE) as the quantity of interest. RMSE measures the distance between the model predictions and the actual physical measurements. In factor prioritisation, using RMSE as the quantity of interest produces results that can guide the determination of which parameters matter the most and least in calibration. Once these are identified, trend identification is used to understand further whether the dependence of RMSE is monotonic or not on the parameters. Interaction quantification, as specified in Ratto et al. (2007) helps with identifiability: parameters associated with high individual contribution are more easily identified than parameters owing their importance to interaction effects. Ratto et al. (2007, p. 1254) point out that "Small main effect but high total effect: here, such a situation flags an influence mainly through interaction, implying lack of identification". Also see Hill and Tiedeman (2007), where they discuss the high composite scaled sensitivities and large parameter correlation coefficients.

We aim to conduct sensitivity analyses that deliver insights on these sensitivity settings simultaneously while keeping computational burden under control. We propose combining a given-data approach for the estimation of global sensitivity measures (Plischke et al., 2013) with the hybrid local-global method DELSA (Rakovec et al., 2014). A given-data approach allows us to exploit the dataset generated for uncertainty quantification to calculate a variety of global sensitivity measures. We then integrate the insights of global methods with the indications yielded by a method capable of extracting information from the partial derivatives dataset.

Table 3.1: Summary of the settings and sensitivity methods used in this chapter.

Setting/Name	Symbol	Equation
Parameter Prioritization		
First-order Sobol'	η_i, S_i	Eq.(2.16), Eq. (2.14)
Borgonovo's δ	δ_i	Eq.(2.22)
Kuiper-based	β^{KU}	Eq.(2.24)
DELSA	S_i^L	Eq.(3.4)[b]
Trend Identification		
Partial Derivatives	$\frac{\partial g}{\partial x_i}$	Eq. (3.8)[b]
Main effect functions	$g_i(x_i)$	Eq.(2.10)
CUSUNORO	$c_i(u)$	Eq.(3.5)
Interaction Quantification		
Sum of First order sensitivity Indices		
Higher order variance-based indices	$S_{i,j}$, ST_i,...	Eq. (2.13), Eq. (2.15)

[b] calculated using local derivatives obtained at distributed points in the parameter space

Our approach addresses each sensitivity setting using multiple sensitivity measures.

This is important because any single sensitivity method refers to a particular aspect of the model output response, has theoretical limitations and, moreover, numerical errors might affect the estimates at finite sample sizes. Thus, by relying on an ensemble of sensitivity measures that can be simultaneously estimated, one increases the robustness of the inference without augmenting the computational burden. For parameter prioritisation, we rely on first-order Sobol' indices [2], on the δ-importance measure (Borgonovo, 2007), on a sensitivity measure based on the Kuiper metric, a modification of the Kolmogorov-Smirnov distance (Kuiper, 1960; Baucells and Borgonovo, 2013) and on a linearized variance index of the DELSA method — see Table 3.1. Note that if the goal of the analysis is model calibration, some sensitivity measures may be preferable. For example, if the objective of the calibration is to minimize the RMSE via gradient/descent method, one may be more interested in looking at the derivative-based sensitivity measures, specially when the uncertain inputs are the same variables on which one is optimizing.

For trend identification, we make use of alternative visualization tools to display results in an intuitive and easy-to-grasp fashion. Because partial derivatives are the natural sensitivity measures for trend identification, we also use them in this work to create derivative scatter plots (D-scatterplot) jointly with the graphs of the global main effect functions of the functional ANOVA expansion and the cumulative sum of normalized reordered output (CUSUNORO) (Plischke, 2010) plots. Indeed, these last two visualization methods do not require partial derivatives, accommodating the case in which the model execution time does not allow the analyst to produce a partial derivatives dataset. Furthermore, we discuss ways to profit from the derivative dataset to analyse the regional contribution of the uncertain parameters. In this case, our goal is to identify whether the importance of a parameter is concentrated in particular ranges of its support (in Ratto et al. (2007), regional sensitivity analysis is associated with a fourth setting, factor mapping). For *interaction quantification*, we use the second-order effects $S_{i,j}$, which are estimated using Polynomial Chaos Expansion (PCE) (Sudret, 2008; Marelli and Sudret, 2015), HDMR (Ziehn and Tomlin, 2009) and LASI (a subroutine based on high dimensional model representations described Section 3.3.2).

We demonstrate the approach by conducting numerical experiments within the well-established hydrologic modelling framework for understanding structural errors (FUSE) (Clark et al., 2008). This framework was the first in the hydrologic sciences to be designed specifically to support consideration of alternative working hypotheses (also called alternative models or multi model analysis) (Clark et al., 2011). We provide results for a medium-sized basin situated in the hilly parts of the Belgian Ardennes (Western Europe). We start with a reference configuration (FUSE-016) and then compare it with other four

[2]Note that one can alos compute the total-order indices for parameter prioritisation. However, their estimation from a given-sample is less convenient than individual global sensitivity indices, thus we consider first-order Sobol indices in this work.

alternative structure configurations while studying the sensitivity of the model RMSE to variations in the parameters. The sensitivity methods allow us to confidently identify the key drivers of RMSE variability across the configurations, to establish whether RMSE is increasing or decreasing in the parameters and to identify the presence of interactions.

The sensitivity analysis insights of this work will be broadly applicable for the next generation modeling frameworks, such as the structure for unifying multiple modeling alternatives (SUMMA) in Clark et al. (2015) and Clark et al. (2015) and the ongoing community-based efforts on parameter regionalization schemes of hydrology/land-surface models (e.g., Mizukami et al., 2017; Samaniego et al., 2017). They also have considerable utility for climate models and other environmental systems (e.g., Mendoza et al., 2015; Cuntz et al., 2015).

The remainder of the chapter is organized as follows. Section 3.2 reviews the sensitivity analysis methods used in correspondence of each setting. Model results are presented in and discussed in Sections 3.4 and 3.5 respectively.

3.2 Methods I: review, definitions and properties

This section is organized as follows. Section 3.2.1 offers a literature review of the use of sensitivity analysis in hydrological modeling. In parameter prioritisation, the analyst aims to identify the most influential input parameters. Sensitivity measures, either global or local, can be used for such identification. Sections 3.2.2, 2.2.2, 2.2.3 and 2.2.4 of Chapter 2 describe the commonly used sensitivity measures in the literature. Section 3.2.3 presents the sensitivity methods associated with the setting trend identification; Section 3.2.4 illustrates the methods for interaction quantification. The notation in this chapter is consistent with that used in Chapter 2.

3.2.1 Concise literature review

In the last decade, the work of many researchers has contributed to make sensitivity analysis a key ingredient of modelling in the hydrologic, environmental and climate change sectors. Pianosi and Wagener (2015), Razavi and Gupta (2016b) and Hill et al. (2016) present recent overviews on environmental and hydrological applications of sensitivity analysis. The works of Saltelli et al. (2012) , Borgonovo and Plischke (2016), Razavi and Gupta (2016b) and Ghanem et al. (2016) provide broad overviews with applications also in other fields. We limit our review to works which are relevant to ours. We start with Pappenberger and Beven (2006), who strongly underline the importance of properly quantifying uncertainty in hydrological modelling, using global sensitivity analysis

methods. The potential of global sensitivity analysis methods in hydrological modelling is further illustrated in Ratto et al. (2007), who performed sensitivity analysis in the context of model calibration and validation, using the generalized likelihood uncertainty estimation (GLUE) method (Beven and Binley, 1992). They perform global sensitivity analysis within the settings of factor prioritisation using variance-based methods and including the elementary effect test of the Morris method (Morris, 1991).

Cloke et al. (2008) are among the first to apply global sensitivity analysis in the context of a complex hydrological model. This work relies on variance-based sensitivity measures and focuses on factor prioritisation. Pappenberger et al. (2008) compare five different methods with a focus on factor prioriziation for the one-dimensional indundation model of the Alzette River basin in Luxembourg. The compared methods are variance-based sensitivity measures (Homma and Saltelli, 1996), Kulback-Leibler entropy (Critchfield and Willard, 1986), the Morris Method, regionalized sensitivity analysis (Young, 1999) and regression (Storlie et al., 2009). The work concludes that alternative methods lead to completely different parameter rankings and it is not possible to draw a firm inference about the key drivers. Tang et al. (2007) compare four sensitivity analysis methods, namely, differential sensitivity, regional sensitivity, analysis of variance and variance-based sensitivity analysis for factor prioritisation of watershed models, concluding that the analysis of variance and Sobol' methods yield more stable rankings. Tang et al. (2007) employ variance-based sensitivity methods for parameter raking in distributed rainfall-runoff models. van Werkhoven et al. (2009) use variance-based sensitivity measures for parameter prioritisation in the context of multi-objective calibration of watershed models. In Dobler and Pappenberger (2013), the authors compare model parameter sensitivity rankings obtained from the Morris method, regional sensitivity analysis and variance-based sensitivity measures calculated using the state-dependent parameter meta-modeling approach — see Ratto and Pagano (2010) for details. Similarly to ours, these works utilize alternative methods for the identification of key drivers of uncertainty. Differently from ours, these works focus mainly on factor prioritisation, and use specific designs for each of the proposed methods — as, indeed, the given-data methodology was not yet fully developed.

To quantify interactions, Rosero et al. (2010) use first and total order variance-based sensitivity indices. Identifying interactions is crucial in hydrological/land surface studies to shed additional light on the underlying phenomena to improve model realism. On the one hand, from the modeler's point of view, parameter interactions may be unwarranted and appear to deteriorate the overall model realism, implying that the model structure can become over-parameterized and parameters unidentifiable (e.g., Rosero et al., 2010; Foglia et al., 2013). Similar findings were recently also discussed by Houle et al. (2017) for two snow models of varying complexities and different degrees of physical realism. On the other hand, parameter interactions may be unavoidable in complex environmental models,

and maybe intrinsic in the physics of the phenomena under investigation (Markstrom et al., 2016). Hill et al. (2016) compare global and differential methods for identifying parameter interactions.

While the works cited so far focus mainly on variance-based sensitivity measures, recently, moment independent methods have become of interest, for their ability to solve some of the limitations associated with variance-based sensitivity measures (moment-independent sensitivity methods derive their name from Borgonovo and Tarantola (2008) who compare them in the context of chemical models with correlated inputs). In particular, first order variance-based sensitivity measures do not possess the nullity-implies independence property (Note that this limitation is addressed by the total-order Sobol indices.). That property is relevant in parameter prioritisation, where we want to be reassured that a null value of a global sensitivity measure implies that the parameter may not play a role in the model. In the environmental sciences, the δ-importance measure of (Borgonovo, 2007; Borgonovo et al., 2012) and the PAWN method of Pianosi and Wagener (2015) have been subject of investigation. Khorashadi Zadeh et al. (2017) provide a recent comparison between variance-based and moment-independent methods. We refer to Section 2.2.3 for details on the common rationale encompassing several global sensitivity measures including variance-based and moment independent methods.

3.2.2 Methods based on semi-local and local sensitivities

[3] The class of differentiation-based (derivative-based) sensitivity measures is regarded as a class of sensitivity measures in a factor fixing setting. Sobol' and Kucherenko (2009) define:

$$\nu_i^{DGSM} = \mathbb{E}_{X_i}\left[\left(\frac{\partial g}{\partial x_i}\right)^2\right]. \tag{3.1}$$

The sensitivity measure in Eq. (3.1) is equal to the average of the square of partial derivatives evaluated at randomized locations in the parameter space. In particular, the DELSA method of Rakovec et al. (2014) combines methodological properties from three other methods: the method of Morris (Morris, 1991), the Sobol' method (Sobol, 2001), and regional sensitivity analysis (Hornberger and Spear, 1981). DELSA uses the local equation for variance estimation V_L (Seber and Wild, 1989; Draper and Smith, 1998; Aster et al., 2013; Lu et al., 2012):

$$V_L = \left(\frac{\partial g}{\partial \boldsymbol{x}}\right)^T (\boldsymbol{X}^T \boldsymbol{\omega} \boldsymbol{X})^{-1} \left(\frac{\partial g}{\partial \boldsymbol{x}}\right), \tag{3.2}$$

[3]The 'semi-local' method refers to a hybrid local-global method.

which linearly propagates the parameter uncertainty expressed by $(\boldsymbol{X}^T\boldsymbol{\omega}\boldsymbol{X})^{-1}$ to obtain the variance of the output — see Appendix A in Rakovec et al. (2014) for additional mathematical details. In Eq.(3.2), we define $\boldsymbol{X}$ as a $k \times k$ identity matrix and $\boldsymbol{\omega}$ is estimated as the reciprocal variance of the uniform distribution from the parameter prior ranges. Then, the total linearized local variance V_L becomes

$$V_L = \sum_{i=1}^{k} \left(\frac{\partial g}{\partial x_i} \right)^2 \sigma_{X_i}^2, \tag{3.3}$$

where $\sigma_{X_i}^2$ is the priori variance of the i-th input. Finally, the DELSA first-order sensitivity measure of the ith parameter is calculated at each sampling point as:

$$S_i^L = \frac{\left(\frac{\partial g}{\partial x_i} \right)^2 \sigma_{X_i}^2}{V_L}. \tag{3.4}$$

Eq. (3.4) is the local fraction of the linearized variance of Y apportioned by X_i. The DELSA indices S_i^L are calculated at randomized locations throughout input space. The analyst can then consider the empirical distribution of these sensitivity measures or any other statistical property for making inference. For instance, the median of the sample of S_i^L is considered in Rakovec et al. (2014) for factor prioritization.

3.2.3 Trend identification: sensitivity measures

In the *trend identification* setting, we address an essential insight about model behavior, the need to understand whether an increase (decrease) in a parameter leads to an increase (decrease) in the model output. The importance of this setting has been appreciated since the seminal work of Samuelson (1941, p. 97):"In order for the analysis to be useful it must provide information concerning the way in which our equilibrium quantities will change as a result of changes in the parameters taken as independent data."As also underlined in Samuelson's work, the appropriate sensitivity measures for this task are signs of partial derivatives. As we shall see, an efficient visualization tool is a derivative scatter plot. If a derivative dataset is not available, we argue that one can make use of the following two methods to still obtain information on trend identification: visualization of the first order terms of the functional ANOVA expansion and use of the CUSUNORO plot. Let us start with the former first-order terms.

The first order functions $g_i(x_i)$ in Eq. (2.10) can be used to obtain information about sign of change. By definition, $g_i(x_i)$ is the conditional expectation of Y given $X_i = x_i$. Thus, $g_i(x_i)$ conveys the average behavior of Y as a function of x_i. Moreover, the first order effect function $g_i(x_i)$ retains the monotonicity of the original input-output mapping

(Beccacece and Borgonovo, 2011). That is, if $g(\mathbf{x})$ is increasing, then all the $g_i(x_i)'$s are increasing. Then, the visualization of the graphs of the first order effect functions provides an indication about the expected trend of Y as a function of x_i.

The CUSUNORO curve for parameter X_i is given for $u \in [0,1]$ by:

$$\mathrm{c}_i(u) = \frac{u}{\sqrt{\mathbb{V}[Y]}}\mathbb{E}\left[Y - \mathbb{E}[Y]\middle|X_i \leq F_{X_i}^{-1}(u)\right] = \frac{1}{\sqrt{\mathbb{V}[Y]}}\int_{-\infty}^{F_{X_i}^{-1}(u)} \mathbb{E}\left[Y - \mathbb{E}[Y]|X_i = x\right] \mathrm{d}x \tag{3.5}$$

By construction, it is a curve with $\mathrm{c}_i(0) = 0$ and $\mathrm{c}_i(1) = 0$. The curve $\mathrm{c}_i(u)$ displays the average mean of the standardized output when the associated parameter is less than a given quantile u. We may therefore speak of a partial mean to the left (given by $F_{X_i}(x)\mathbb{E}[Y - \mathbb{E}[Y]|X_i \leq x]$ in contrast to the conditional mean to the left given by $\mathbb{E}[Y - \mathbb{E}[Y]|X_i \leq x]$) and note that due to standardization, this mean to the left and the corresponding mean to the right add up to zero (Plischke, 2010). If the model is an increasing function of X_i, then the partial mean to the left is always lagging behind the global mean. Therefore, the CUSUNORO curve is negative for all values of u. Conversely, if the model is decreasing in X_i, then the CUSUNORO curve is positive for all values of u. It can also be proven that if there exists a linear regression curve with respect to the estimated cdf of the parameters

$$\mathbb{E}[Y - \mathbb{E}[Y]|\hat{F}_{X_i}(X_i) = u] = \alpha(u - \tfrac{1}{2}) \tag{3.6}$$

the CUSUNORO curve has a local extremum at $u_0 = \frac{1}{2}$. Hence any extreme value not located in the center of the CUSUNORO plot shows a nonlinear dependence between Y and X_i.

3.2.4 Interaction quantification: sensitivity measures

In an interaction quantification setting, we are interested in understanding whether the model response is additive or not. If the response is additive, then the variation of the output is the direct sum of the individual effects of the variations in the parameters. Herein, we aim at studying interactions while remaining in a given-data frame. As underlined in Saltelli et al. (2000) the quantity

$$S^{\text{Interaction}} = 1 - \sum_{i=1}^{k} S_i \tag{3.7}$$

can be considered as an indicator of the percentage of the model output variation apportioned by interactions. Because first order variance-based sensitivity indices can be

estimated from a given-data frame, this quantity $S^{\text{Interaction}}$ is also delivered by a given-data approach. Then, if the sum of S_i (or η_i) is close to 1, we are informed that interactions provide a limited contribution to the model output variation, so that the model response can be regarded as additive. Otherwise, further investigation on the nature of interactions is needed. Several methods are available. For instance, one can start investigating the effects of the interactions of all pairs, through linear inferential measures (Hill and Tiedeman, 2007; Hill et al., 2016). Herein, we rely on second order Sobol' sensitivity indices $S_{i,j}$ - see Section 2.2.2 of Chapter 2 for the mathematical definitions.

Alternative ways are available for estimating second order Sobol' sensitivity indices directly from the Monte Carlo sample. We employ here two methods based on the high dimensional model representation (HDMR) theory, namely, HDMR (Ziehn and Tomlin, 2009) and LASI, see Section 3.3.2 for further details. We compare the two HDMR based estimations to the estimation method based on PCE (Sudret, 2008; Marelli and Sudret, 2015).

3.3 Methods II: numerical estimation and graphical representation

3.3.1 Given-data and derivative estimation

We consider that a team of hydrologists has developed a complex hydrological code. Performing a global uncertainty quantification has produced a sample of input-output realizations. Here we focus on using this dataset containing n realizations of the input parameters and the corresponding model output realizations to extract insights on parameter prioritisation, parameter fixing, trend and interaction quantification. This way of proceeding, with sensitivity measures estimated directly from the sample generated for an uncertainty quantification and without a specific design is either called one-sample estimation or given-data estimation — see Strong et al. (2012), Strong and Oakley (2013), Plischke et al. (2013), and Borgonovo et al. (2016) for detailed accounts and the related theory. A cursory review of the principles of this estimation technique is presented in Section 2.2.5 of Chapter 2 and Section 4.2.1 of Chapter 4.

Because n is unavoidably finite, best practices recommend to assess the error bounds in the estimates. In this chapter, we use bootstrapping and the *bias-reducing bootstrap estimator* of Efron and Gong (1983). The distribution of the bootstrap estimator provides a convenient way to assess the error bound and does not require additional model runs. Thus, in a given-data framework, the estimation cost of the sensitivity measures of interest

is equal to n model runs.

The estimation of partial derivatives is a widely studied subject in the literature. In the present work, we use the derivatives dataset generated by Rakovec et al. (2014), who employ Newton's ratio

$$\widehat{\frac{\partial g(\mathbf{x})}{\partial x_i}} = \frac{g(x_i + h, \mathbf{x}^0_{-i}) - g(x_i, \mathbf{x}^0_{-i})}{h}. \tag{3.8}$$

for the computation of partial derivatives. Conveniently, small values of h is used. With Eq. (3.8), the computational cost for estimating all k first order partial derivatives at a given location is $k + 1$ model runs. When this estimation is randomized in the model parameter space, the computational cost becomes $n(k + 1)$. We do not indulge further in the discussion of the computation of partial derivatives as this is a widely discussed subject. However, we refer to Griewank and Walther (2008); Neidinger (2010) and Peckham et al. (2016), among others, for additional details.

3.3.2 Estimation of first order and second order indices using harmonic functions

The main idea for the estimation of variance-based sensitivity indices from given data is their interpretation as nonlinear goodness-of-fit measures (R^2) for suitable regression models. We therefore populate a design matrix $D = [\Psi_1(u) \dots \Psi_M(u)]_{n\times M}$ with the harmonic feature maps $\Psi_m(u) = \sqrt{\frac{2}{n}}\cos(\pi u m)$ where $u = (\frac{2\ell-1}{2n})_{\ell=1,\dots,n}$. Here n is the sample size and M is the maximal higher harmonic to consider (usually $M = 4$ to 8).

The dependence on the factor of interest i enters the regression model by reordering y using x_i as key: there exists a permutation π^i such that $x_{\pi^i(\ell)i} \leq x_{\pi^i(\ell+1)i}$ for all $\ell = 1, \dots, n-1$.

The coefficients from the least squares solution $\beta(i) = D(D^*D)^{-1}D^*y_{\pi^i}$ yield the first order estimate $\hat{S}_i = \frac{\sum_{m=1}^{M}\beta_m^2(i)}{(n-M)\hat{\sigma}_Y^2}$.

The first order ANOVA functions are given by the graphs linking points $(x_{\pi^i(\cdot)i}, \hat{y}_i)$ with $\hat{y}_i = \bar{y} + D^*\beta(i), i = 1 \dots k$. As $-\Psi_1(u) \approx \sqrt{\frac{2}{n}}(2u - 1)$ the opposite of the first coefficient may serve as a monotonic trend indicator.

For second order effects, we mix two design matrices. In order to do so, the associated reverse permutations on the design matrices is needed, instead of the permutation on y. Let ψ^i such that $\psi^i(\pi^i(\ell)) = \ell$ denote the reverse look-up permutation and D_{ψ^i} be the design matrix with permuted rows. Then the two-dimensional design matrix is $D_{ij} = D_{\psi^i} \odot D_{\psi^j}$ where $\odot$ multiplies each column of the first matrix with each column of the second matrix, e.g., for $M = 2$ we have $D_{ij} = [\Psi_1(u_{\psi^i}) \cdot$

$\Psi_1(u_{\psi^j})\,\Psi_1(u_{\psi^i}) \cdot \Psi_2(u_{\psi^j})\,\Psi_2(u_{\psi^i}) \cdot \Psi_1(u_{\psi^j})\,\Psi_2(u_{\psi^i}) \cdot \Psi_2(u_{\psi^j})]$. The regression coefficients are $\gamma(i,j) = D_{ij}(D_{ij}^* D_{ij})^{-1} D_{ij}^* y$ and the estimates of the second order indices are given by $\hat{S}_{ij} = \frac{\sum_{m=1}^{M^2} \gamma_m^2(i,j)}{(n-M)^2 \hat{\sigma}_Y^2}$.

3.3.3 Graphical methods and visualization of sensitivity results

Partial derivatives are used for DELSA and for *trend identification*. This setting is particularly amenable to a graphical representation of results, and visualization can serve as a nice bridge between the analyst and the decision maker. We first introduce the construction of *D-scatterplots*. In a D-scatterplot one forms a scatter plot of x_i and the values of the partial derivatives of the model output estimated at x_i . If the model is monotonically increasing in x_i then the dots of the scatter plot are located only in the first and second quadrants; if y is monotonically decreasing in x_i, then the dots are located only in the third and fourth quadrants.

Several methods and subroutines are available to estimate and plot fist-order effect functions from a given-dataset. For instance, the GUI-HDMR Matlab code of Ziehn and Tomlin (2009) or the routine based on smoothing spline regression of Ratto and Pagano (2010). Here we make use of the COSI subroutine provided by Plischke (2012), where mathematical details are provided. The COSI method obtains the graph of the estimated first order terms $g_i(x_i)$ from a harmonic regression.

A further way to obtain indications about trend from the uncertainty quantification sample is to build a CUSUNORO plot. A CUSUNORO curve for model input X_i can be constructed through the following simple steps:

1. Permute the input parameter of interest $(x_1, \ldots, x_n)$ (the input superscripts i are omitted) in ascending order (order statistics), so we have $x_{\pi(j)} < x_{\pi(j+1)}$, $j = 1, \ldots, n-1$ using the index permutation π, and corresponding re-ordered model output $\left\{y_{\pi(j)}\right\}_{j=1}^{n}$.

2. Calculate the scaled cumulative sums of centered re-ordered output as follows:

$$z(j) = \frac{\frac{1}{n}\sum_{m=1}^{j}\left(y_{\pi(m)} - \bar{y}\right)}{\sqrt{\frac{1}{n-1}\sum_{n=1}^{n}\left(y_n - \bar{y}\right)^2}}. \tag{3.9}$$

3. Plot the pairs $\left(\frac{j}{n}, z(j)\right), j = 1, \ldots, n$ with the convention imposing $z(0) = 0$.

The above procedure is repeated for each parameter and the CUSUNORO curves are plotted then in the same graph, yielding the CUSUNORO plot. From this plot, we obtain insights concerning trend as discussed above.

Interaction quantification can be very computationally demanding. The first step is to quantify the sum of first order variance-based sensitivity measures. To calculate higher order Sobol' indices, several strategies are available. In a brute force approach, the estimator strictly follows the definition of these sensitivity measures. This requires a double-loop Monte Carlo simulation approach. In this case, the cost is equal to $\binom{k}{2}n$ model runs. The computing time may render the estimation impractical, especially if the model evaluations are time-consuming or if k is large. Researchers have therefore investigated computational reduction strategies since the late 90's. For example, Saltelli (2002a) presents a method in which all first, total and non-normalized second order terms $g_{i,j}$ can be estimated at a cost of $n(2k+2)$ model runs. In this work, we estimate all interactions at a cost of n model runs by building a metamodel to replace the original model, namely, PCE, HDMR and LASI. A dedicated review of meta-modelling techniques in the hydrological sciences is discussed in Razavi et al. (2012).

3.4 Application

3.4.1 Hydrological framework

The ensemble of sensitivity methods described in the aforementioned sections is executed using a set of models developed to simulate a medium-sized catchment (Lasnenville, 200 km^2) located in the Belgian Ardennes (Western Europe). The maritime climate can be classified as rain-dominated with irregular snow in the winter. The runoff regime is highly variable with low summer discharges and high winter discharges. The annual precipitation yields around 1 000 mm and the mean annual air-temperature is 7.5°C. Mixed-forest and agricultural areas represent the two dominant land cover classes (Rakovec et al., 2012).

Five models are developed using the Framework for Understanding Structural Errors (FUSE), a well-established modular framework, which enables the construction of a suite of hydrological models to rigorously implement and evaluate hydrological theories (Clark et al., 2008, 2011). The ability of a model to adequately approximate dominant hydrological processes depends on (1) the choice of state variable in the unsaturated and saturated zones, and (2) the choice of flux equations describing the surface runoff, vertical drainage between soil layers, baseflow and evapotranspiration (Clark et al., 2008).

The vertical dimension of all models is discretized into two reservoirs: the unsaturated reservoir above the water table (often referred to as soil moisture storage) and the saturated one below the water table (also known as groundwater storage). The outflow of the two model reservoirs constitutes the total simulated river flow, which is often also called river discharge and/or streamflow. The model output g used to assess model sensitivity of this study is an aggregated metric over T daily time steps. The metric g is defined

to quantify discrepancies between the model and reality (real-world measurements/observations)as the root-mean-square-error (RMSE) between the simulated streamflow (q_{sim}) and the observed streamflow (q_{obs}):

$$g = \text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(q_{obs,t} - q_{sim,t})^2}. \tag{3.10}$$

The FUSE-016 configuration has a "single-layer" architecture for the unsaturated zone, which does not allow for vertical variability in soil moisture. Evapotranspiration is restricted to the upper unsaturated zone, and is a linear function of the storage between the wilting point and the field capacity. The FUSE-016 does not allow any vertical drainage when the saturation is below field capacity. FUSE-016 has a single nonlinear groundwater reservoir of unlimited size. The surface runoff is conceptualized using the "ARNO/VIC" parameterization, and the routing schemes employ the time delay function using a gamma distribution. The FUSE-014 and FUSE-160 models extend the FUSE-016 configuration by alternative evapotranspiration processes from the unsaturated zone, which is represented by two cascading reservoirs. The FUSE-072 model enables for vertical drainage through a non-linear function, which is the only difference with respect to FUSE-016. The FUSE-170 configuration addresses alternative representations of the base flow parameterizations with respect to FUSE-016 by employing two linear groundwater storages. We refer to Clark et al. (2011) for further details.

The number of parameters for the five FUSE models ranges between 11 and 14 (Clark et al., 2008; Rakovec et al., 2014). The parameters are summarized in Table 3.2. The parameters can not be directly measured in nature with sufficient accuracy, and are location-specific based on the regional climate and physiographic basin properties.

This study makes use of the model simulations at daily time steps presented by Rakovec et al. (2014) for a 10-year period from 1 October 1998 to 30 September 2008. The parameter ranges applied in this study are slightly adjusted from Clark et al. (2011). Note that the sample size n of this study is 9 548, which represents the number of base model runs for the FUSE-016 model. The difference from the 10 000 runs used in Rakovec et al. (2014) originates from revising the lower parameter bound for TIMEDELAY from 0.01 to 0.1.

In the reminder of this section, we focus the presentation on results for the FUSE-016 configuration, while the four alternative models FUSE-014, FUSE-160, FUSE-072, and FUSE-170 are used to assess the robustness of the parameter sensitivity analysis methods for alternative model structures. Results for these models are described in Section 3.4.5. Besides, the model outputs, subroutines and datasets can be obtained from the publicly available repository: `https://github.com/rakovec/Making_the_Most_out_of_a_HM_Dataset`.

Table 3.2: The ranges for parameters of five FUSE models are as defined in Rakovec et al. (2014), except for the lower bound of TIMEDELAY (shown in bold). Smaller values of TIMEDELAY produce unreasonable results given the one-day time step of the model. Note that the parameters 1–11 belong to the FUSE-016 configuration. Parameters 12-18 belong to extra processes incorporated within alternative model structures of FUSE-014, FUSE-160, FUSE-072, and FUSE-170.

No.	Parameter name	Description	Units	Lower limit	Upper limit
1	MAXWATR_1	Maximum storage in the unsaturated zone	mm	50	500
2	MAXWATR_2	Maximum storage in the saturated zone	mm	25	250
3	FRACTEN	Fraction total storage as tension storage	–	0.05	0.95
4	PERCRTE	Vertical drainage rate	mm/day	0.01	1000
5	PERCEXP	Vertical drainage exponent	–	1	20
6	BASERTE	Baseflow depletion rate	mm/day	0.001	1000
7	QB_POWR	Baseflow exponent	–	1	10
8	AXV_BEXP	ARNO/VIC “b” exponent for surface runoff	–	0.001	3
9	LOGLAMB	Mean of the log-transformed TI[a] distibution	m	5	10
10	TISHAPE	Shape parameter for TI[a] distribution	–	2	5
11	TIMEDELAY	Routing parameter (time delay in runoff)	day	**0.1**	2
12	FRCHZNE	Fraction of tension storage in the primary zone (unsaturated zone)	–	0.05	0.95
13	FPRIMQB	Fraction of free storage in the primary reservoir (saturated zone)	–	0.05	0.95
14	RTFRAC1	Fraction of roots in the upper soil layer	–	0.05	0.95
15	PERCFRAC	Fraction of drainage to tension storage in the lower layer	–	0.05	0.95
16	FRACLOWZ	Fraction of soil excess to lower zone	–	0.05	0.95
17	QBRATE_2A	Baseflow depletion rate for the primary reservoir	day^{-1}	0.001	0.25
18	QBRATE_2B	Baseflow depletion rate for the secondary reservoir	day^{-1}	0.001	0.25

[a] TI: topographic index

3.4.2 Parameter prioritisation results

With the aim of identifying the most important parameters, we use an ensemble of sensitivity indices combining indications from variance-based, density-based and cdf-based global sensitivity measures. Specifically, using the given data estimators described in Section 3.3, we estimate first-order Sobol' indices η_i, Borgonovo's δ_i, and the Kuiper index β_i^{KU}.

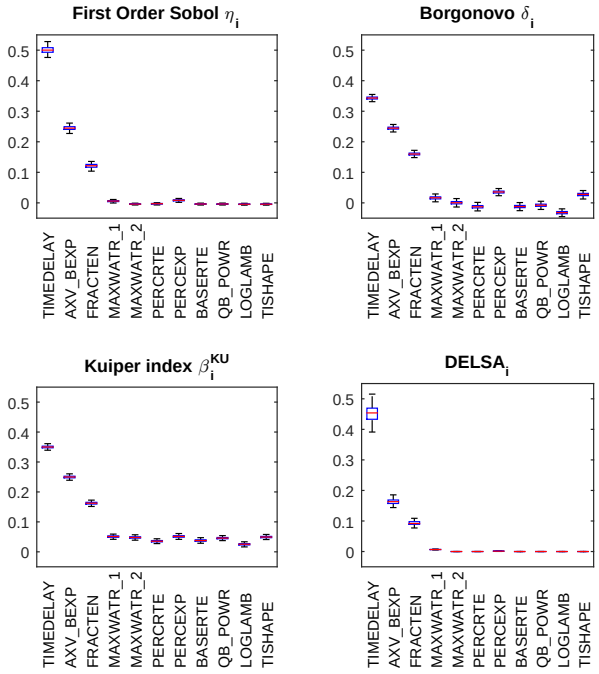


Figure 3.1: Factor prioritisation using four methods. Boxplots for the bootstrap estimates for three global sensitivity measures $\left(\eta_i, \delta_i, \beta_i^{KU}\right)$, with 500 bootstrap replicates ($C = 9548$). Similarly, the bootstrap median of DELSA is presented in the fourth graph. All sensitivity indices agree in suggesting TIMEDELAY, AXV_BEXP and FRACTEN as key uncertainty drivers.

Figure 3.1 shows boxplots of the bootstrap estimates of these three global sensitivity measures, with a bootstrap sample size $B = 500$. All three approaches rank TIMEDELAY, AXV_BEXP and FRACTEN as the most influential parameters. The first two parameters directly influence the dynamics of simulated streamflow, in particular its timing and magnitude (TIMEDELAY), and the partitioning of incoming precipitation into quickly responding surface runoff and slow baseflow components (AXV_BEXP). Their role, therefore, explains the direct and strong influence on the RMSE in Eq. (3.10), which is derived directly from the simulated streamflow. The third most influential parameter (FRACTEN) has a direct effect on the soil moisture dynamics. It quantifies tension storage as a non-linear function of the total storage in the unsaturated zone. FRACTEN closely controls the magnitude of evapotranspiration processes, i.e., the return of incoming precipitation back to the atmosphere, and it also indirectly affects the magnitude of the total modelled streamflow. Overall, the importance of the three key parameters is identified clearly and consistently, which is shown by the narrow and not overlapping bootstrap uncertainty bounds. Furthermore, Figure 3.1 includes the parameter ranking

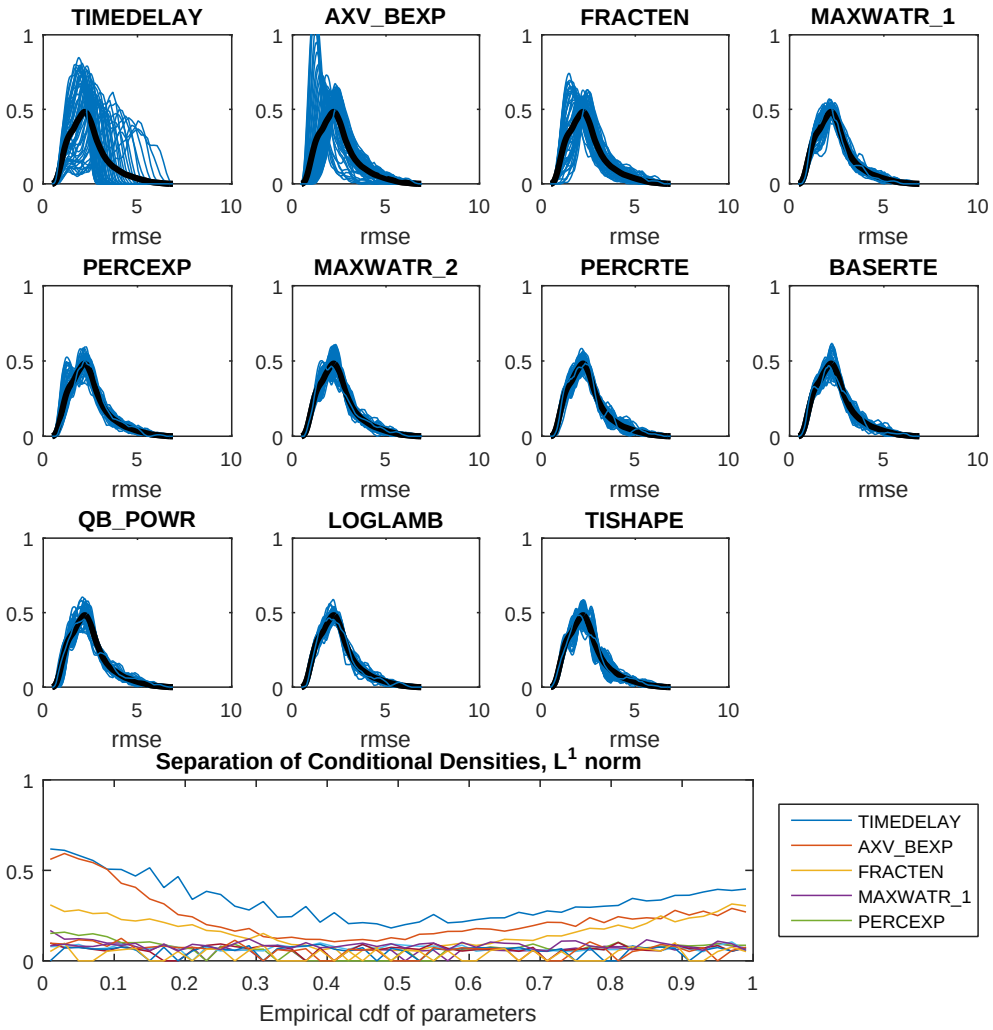


Figure 3.2: Visualization of estimated conditional and unconditional model output densities for the FUSE-016 configuration. The bold black line in each of the first eleven panels represents the estimated unconditional RMSE density $\hat{f}_Y(y)$. The blue lines in each graph represent the estimated bin conditional distributions $\hat{f}_{Y|X_i \in \mathcal{X}_i^m}$. Densities are estimated using kernel smoothing method. The first three panels show that fixing TIMEDELAY, AXV_BEXP and FRACTEN leads to conditional distributions that may largely deviate from the unconditional distribution (black line). The remaining panels show lesser deviations associated with fixing the remaining parameters. The last panel displays the estimated separations at each value of X_i using Eq. (2.19). That is, each line is an approximation of the curve $\zeta_i(x_i) = \frac{1}{2}\int_{\mathcal{Y}} \mid f_Y(y) - f_{Y|X_i=x_i}(y) \mid \mathrm{d}y$. Results in this panel show that the separations of TIMEDELAY, AXV_BEXP and FRACTEN are systematically higher than the remaining separations.

obtained using the median of DELSA index S_i^L (we use $n = 9548$ model runs). The results are in clear agreement with the results produced by the three global sensitivity measures estimated directly on the uncertainty quantification sample.

We provide a visual complement to these results in Figure 3.2. As we have seen in Section 2.2.3 of Chapter 2, global sensitivity measures quantify the discrepancy between the unconditional and conditional model output distributions over the entire support. A large difference between $f_Y(y)$ and $f_{Y|X_i}(y)$ implies that the model output is sensitive to X_i. In a given-data estimation we can profit of our knowledge of the estimated $f_Y(y)$ and $f_{Y|X_i \in \mathcal{X}_i^m}$ to visualize this effect. Consider the first graph in Figure 3.2. The black thick line refers to the unconditional density $f_Y(y)$. Each of the blue lines is a conditional density $f_{Y|X_i \in \mathcal{X}_i^m}$, i.e., it represents the conditional RMSE distribution given that X_i belongs to a certain bin. We can visualize whether the conditional lines (in blue) are close to, or depart from the unconditional black line. In the first three graphs, we see notable departures. Thus, we would expected the RMSE to be sensitive to the parameters in the first three graphs. These parameters are indeed the three most relevant parameters, as identified by all global sensitivity measures, namely TIMEDELAY, AXV_BEXP and FRACTEN. The remaining graphs display colored curves much closer to the black line, signalling that the parameters are less relevant. Besides, Figure 3.2 shows how the output distribution is affetcted. For instance, for the first parameter (the first plot of Figure 3.2), the influence is evenly spread-out over the entire output support, while the second parameter tends to influence the lower values of the output. One can also see that both the first and second parameters shift the unconditional distribution to the left, while the third one has a more symmetric influence.

The last graph in the lower panel of Figure 3.2 visualizes the inner statistic of a given-data estimation, i.e., it is a plot of the curves $\zeta_i(x_i) = \zeta(f_Y(y), f_{Y|X_i=x_i})$ which represent the separation between the unconditional and conditional distribution using the L^1-distance. We can see that the L^1 separations associated with TIMEDELAY, AXV_BEXP and FRACTEN are larger at any value of X_i than the separations associated with the remaining parameters.

To corroborate these results, we use the median statistic of the DELSA in Eq.(3.4) (Figure 3.1). This graph identifies LOGLAMB, PERCRTE, BASERTE as the least relevant parameters, which is consistent with the ranking of global sensitivity measures. Thus, for the RMSE of FUSE-016, we can not only identify the key drivers, but are also able to figure out the parameters that can be fixed at their base case because they are less influential.

Figure 3.3 displays the derivative-based sensitivities, with the goal of identifying important parameter regions, in a factor mapping setting. For illustration purposes, we limit our attention to four parameters (columns in Figure 3.3). Figure 3.3 row (a) presents the S_i^L sensitivities on the horizontal axis and the model output value on the y-axis, as per Rakovec et al. (2014). These results show a crucial difference in the contribution of TIMEDELAY, and AXV_BEXP, the parameters associated with the largest sensitivities. TIMEDELAY is most important in the subset of parameter values associated with higher RMSE, i.e., where the performance of the model is poorer. Conversely, AXV_BEXP is important in regions of lower RMSE, i.e., where the model has a better prediction capability. These considerations show that using information

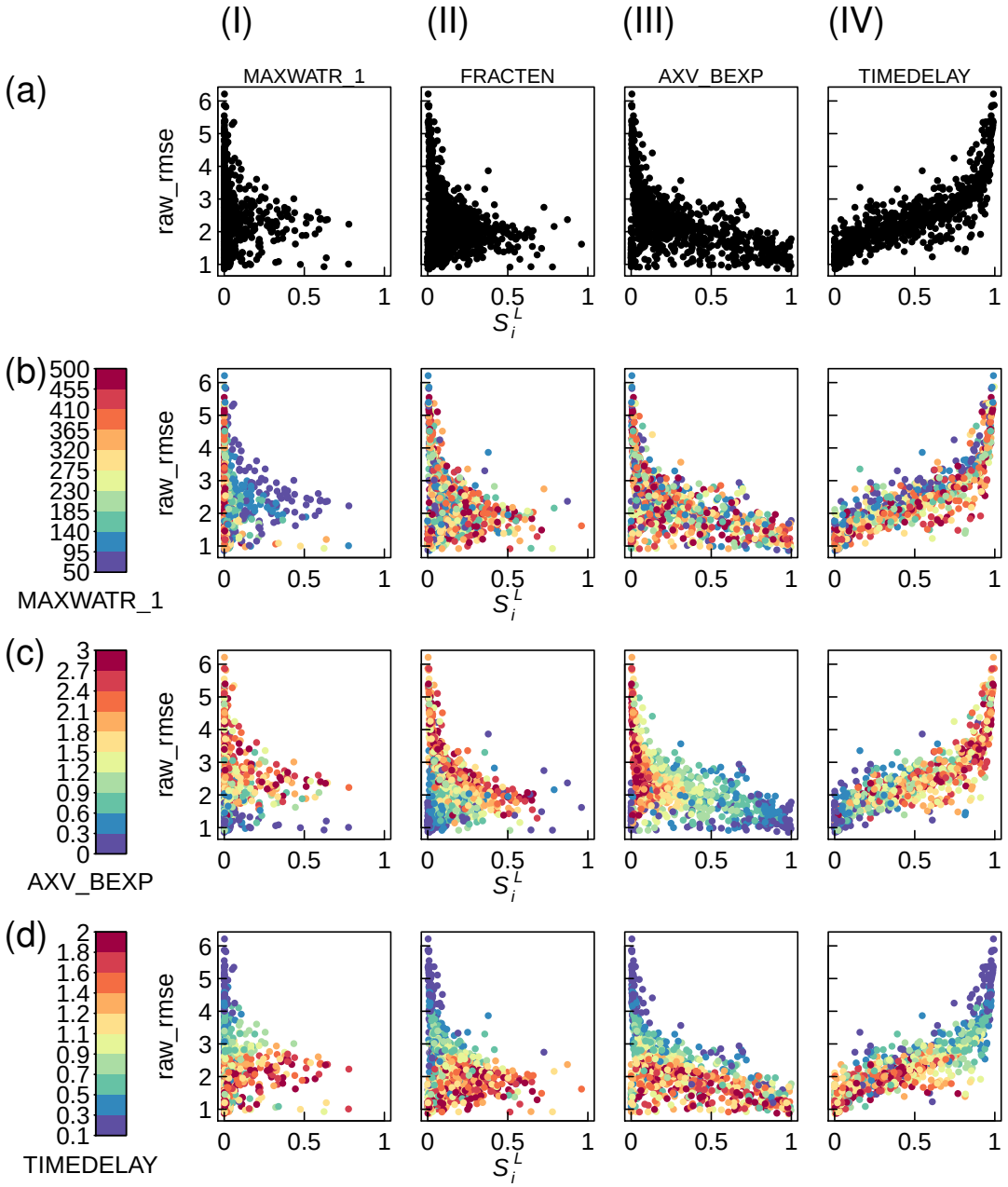


Figure 3.3: Prioritization using DELSA. DELSA results showing parameter importance, measured using first-order metric S_i^L, plotted against the model output root mean-squared error (RMSE). Each dot represents a scaled local sensitivity calculated for one input point. 9 548 dots are shown in each figure. The RMSE for each dot is the same in each figure; the S_i^L value changes. (a) Black and white figures emphasize the position of the dots for the parameters in columns (I-IV). (b-d) The dots are colored based on the value of the parameter listed below the color bar.

coming from derivatives in a regionalized DELSA setting allows the analyst to delve deeper into how each parameter contributes to the RMSE variability. This leads to insights that enrich and complement the information in Figure 3.1.
For this model, the partitioning of incoming precipitation into fast and slow flow components governed by AXV_BEXP may be more important than the routing dynamics characterized by the TIMEDELAY parameters, if the focus is on the "acceptable" model performance-simulations. Additionally, although FRACTEN and MAXWATR_1 exhibit considerably less pronounced importance, some parameter combinations yield $S_i^L > 0.5$, and some of these are very good fitting models based on RMSE.

Figures 3.3b-d yield additional insights. *Random color scatter* indicates that the value of parameter importance (measured here using S_i^L) and model fit (measured here by RMSE) are unrelated to the value of the parameter (row b-d). Thus, the value of the MAXWATR_1 parameter (row b) is mostly inconsequential to the results shown for FRACTEN and AXV_BEXP (columns II and III). The only possible pattern is that the worst-fitting models appear to be dominated by small values of MAXWATR_1.

If the color of dots changes vertically in the plots, then the model fit depends on the parameter value. For instance, Figure 3.3, panel IV-b shows that for any parameter importance level (for parameter TIMEDELAY), poorer fitting models are dominated by larger values of the MAXWATR_1 parameter. Figure 3.3 panel I-c, and all of row d show vertical patterns in dot color.

If the color of dots changes horizontally in the plots, then importance of the parameter depends on the value of the parameter. For example. Figure 3.3 panel I-b shows that nearly all models with large values of MAXWATR_1 are insensitive to the MAXWATR_1 parameter. Figure 3.3 panel IV-d shows that large sensitivities for the TIMEDELAY parameter (the most important parameter) are related to values of the TIMEDELAY parameter smaller than about one day. The time step of the model used is one day, and this suggested that evaluation of models with these small TIMEDELAY values are worth considering closely.

Finally, a note on computational cost. Keeping computational burden under control is essential to reassure the analyst about the feasibility of the analysis. In our case, the available sample consists of about $n = 10,000$ realizations. The relevant question for this work is whether we can consider that estimates are stable at this sample size. Moreover, from the available sample we can also address the question of what is the minimum sample size at which stable estimates of the sensitivity measures are registered for this model. To answer these questions it is enough to display the sequence of the sensitivity measures' estimates as the sample size increases. Figure 3.4 reports such sequences for the four sensitivity measures of Figure 3.1. On the horizontal axis in Figure 3.4 the sample size ranges from $n = 200$ to 9548. The first observation is that all estimates are stable at $n = 9548$. By the consistency theorem of the given data methodology (Borgonovo et al., 2016), we can then trust numerical estimates from this sample. Regarding the minimal sample sizes, we observe that the rank of parameters revealed by DELSA median estimates becomes stable at about $n = 600$ with the most and least important parameters identified even at the 200 model run level. At $n = 600$, also the given-data estimates of Sobol'

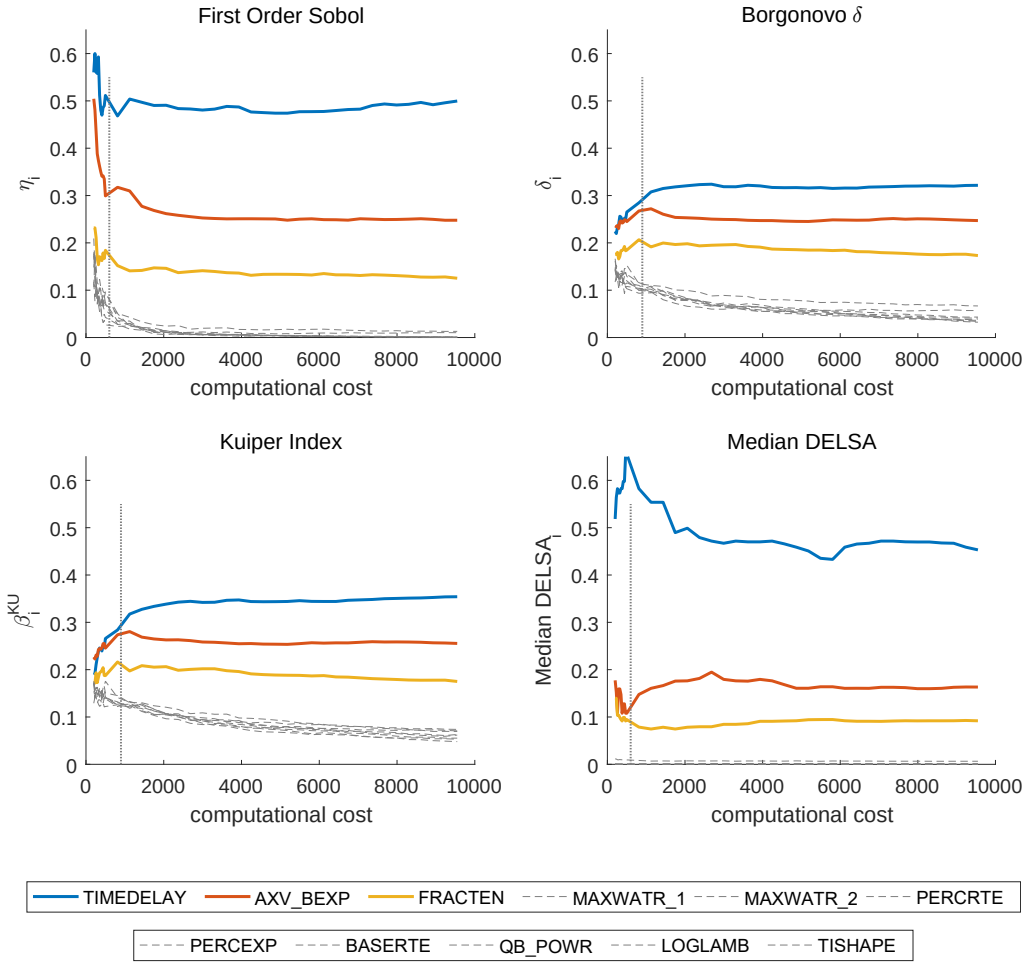


Figure 3.4: Factor prioritisation measures of $\left(\eta_i, \delta_i, \beta_i^{KU}, S_i^L\right)$, as computational cost C measures in number of model runs increases from 200 to 9548. We observe that the available sample of size 9548 from previous studies (Rakovec et al., 2014) leads to a consistent identification of the most important parameters with all sensitivity measures.

sensitivity indices become stable. Estimates of moment independent sensitivity measures stabilize at a sample size of about 900. Thus, for the current analysis a computational cost of about $n = 1,000$ model runs could be considered sufficient to identify the rank of parameters.

3.4.3 Trend identification results

For trend identification, we consider two situations: in the first case, the available dataset comprises partial derivatives; in the second case, only the sample of input-output realizations are available.

In the first case, the sign of the partial derivatives immediately identifies direction of change. Figure 3.5 shows the D-scatterplot for the FUSE-016 model, plotting the derivatives made

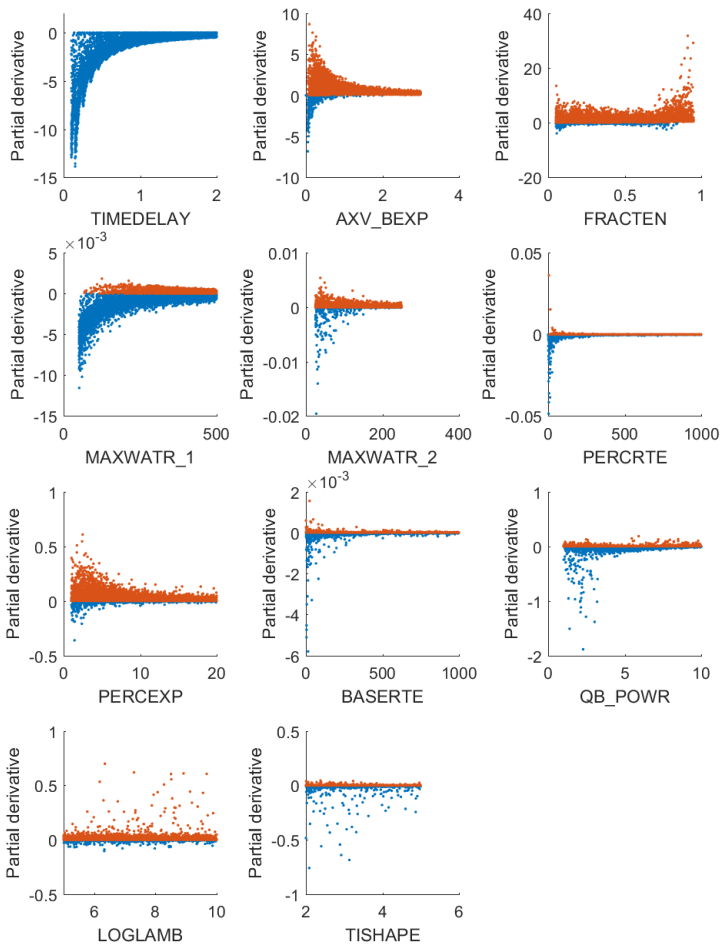


Figure 3.5: Trend identification using D-scatter plots for the eleven parameters of the FUSE-016 model. The vertical axis in each panel displays the estimated derivative of the RMSE with respect to the corresponding parameter evaluated at sampled points . Note: the axis are not standardized, because the goal of the plot is to indicate sign (trend) and not importance.

available by the DELSA method. Light color dots (red) refer to positive values, darker dots (blue) to negative values. The upper left panel of Figure 3.5 shows that the derivatives of RMSE with respect to TIMEDELAY estimated at several locations are negative. Thus, we expect that an increase in TIMEDELAY has a decreasing effect on the RMSE. The upper middle panel of Figure 3.5 plots the derivatives of RMSE with respect to AXV_BEXP. We observe both positive and negative values, which implies that an increase in AXV_BEXP does not necessarily lead to an RMSE increase. Specifically, if we look at AXV_BEXP region $[0, 1]$, we observe both light(red) and dark(blue) dots in the graph (upper middle panel in Figure 3.5). For values of AXV_BEXP greater than unity, however, we observe mainly light dots, which indicates that the effect of an increase in AXV_BEXP leads to an increase in RMSE. The upper right panel of Figure 3.5 presents the derivatives of RMSE with respect to the parameter FRACTEN. Again, the existence of both positive and negative values implies non-monotonicity. However, we observe a majority of positive derivatives (light dots (red)), which indicates an on-average

positive effect. The remaining panels of Figure 3.5 show the existence of both positive and negative values of the estimated derivatives for the remaining parameters.

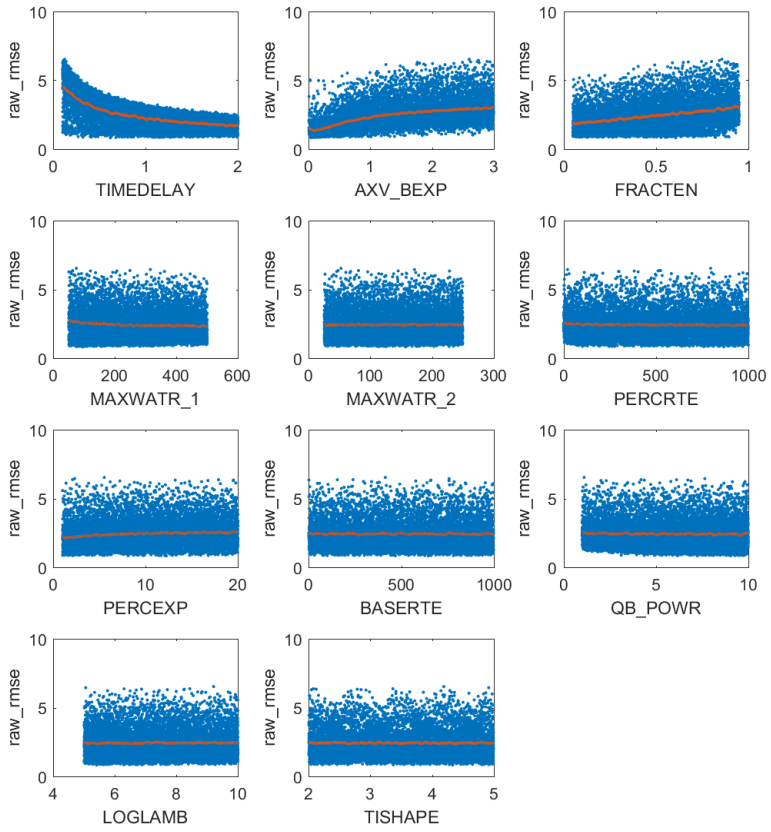


Figure 3.6: Trend identification using first order effect $g_i(x_i)$ for the FUSE-016 model. The estimated curves $g_i(x_i)$ using COSI subroutines (red lines) and input-output scatter plots (blue dots) are shown.

Insights on direction of change can also be directly obtained from the original dataset. A first way is to plot the first order effect functions in Figure 3.6, where we present the COSI curves (in red) together with the input-output scatter plots. One can observe that TIMEDELAY shows a on-average non-linear decreasing effect on the RMSE, while, in general, AXV_BEXP and FRACTEN present an ascending trend. In particular, the trend of AXV_BEXP goes from decreasing to increasing on its support $[0, 1/2]$, which is consistent with the result of D-scatterplot, where a non-monotonic effect is observed in the same region. Besides, PERCEXP shows slightly increasing effect. For the remaining parameters, there is no strong evidence of decreasing or increasing first order effect.

Figure 3.7 illustrates the CUSUNORO plot of FUSE-016 data. Each curve refers to a given parameter. Curves above the horizontal zero line signal a decreasing effect, curves below the horizontal zero line suggest the opposite. Parameter TIMEDELAY (-o- curve) is associated with the CUSUNORO curve that shows the highest peak above the zero horizontal axis, and is therefore the parameter with the strongest negative impact on the RMSE. This is in accordance with the information provided by the first graph in both Figures 3.5 and 3.6. Similarly,

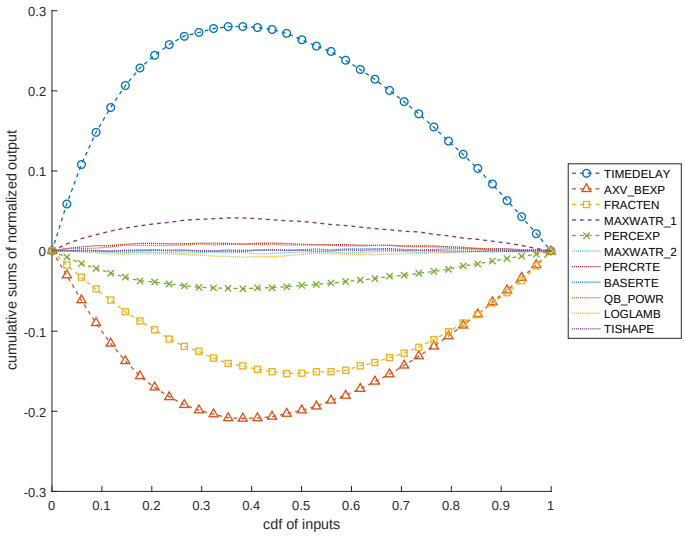


Figure 3.7: Trend identification using the CUSUNORO plot for the FUSE-016 model. Curves above the zero horizontal line indicate a decreasing effect (model output decreases with increasing parameter values). Curves below the horizontal axis show an increasing effect (output increases with increasing parameter values). Curves aligned with the zero horizontal line show a negligible effect.

parameter MAXWATR_1 (−−curve) has a negative effect. Conversely, parameters AXV_BEXP (-△- curve) and FRACTEN (-□- curve) have an increasing effect on the RMSE. Besides, the magnitudes of the deviations from the zero horizontal line can be used to infer information about the strength of the impact. Figure 3.7 indicates TIMEDELAY, AXV_BEXP and FRACTEN as the three most relevant parameters, in accordance with previous findings. Furthermore, the vertical asymmetry of the CUSUNORO curve implies the non-linearity of the first order effect. For instance, TIMEDELAY and AXV_BEXP are slightly asymmetric to the right (have steeper left parts), which implies that we can expect non-linear first order effects. This result is, again, consistent with the graphs of the first order effect functions in Figure 3.6.

3.4.4 Interaction quantification results

A well established method for the identification of interactions is to check the sum of the first order Sobol' indices. From Figure 3.1, we observe that, the sum of first order Sobol' indices is about 90%, indicating that interactions have limited relevance. Thus, the RMSE can be considered a nearly additive function of the parameters over the ranges of interest. However, to investigate further, we study second order interactions, calculating the second order Sobol' indices $S_{i,j}$. We compare three different estimation methods: PCE emulation module in UQ_lab (Sudret, 2008; Marelli and Sudret, 2015), HDMR (Ziehn and Tomlin, 2009) and LASI (see Section 3.3.2). The HDMR and LASI allow to estimate the second order indices directly from the

available 9548 model input-output realizations. The PCE subroutine is trained on a subsample of size 2000 (this is the largest at which calculations can be performed on the available pc without encountering an out-of-memory error). We calculate all second order interactions but are showing only the largest ones. Figure 3.8 (a) illustrates the second order interactions associated with the five largest values of $S_{i,j}$.

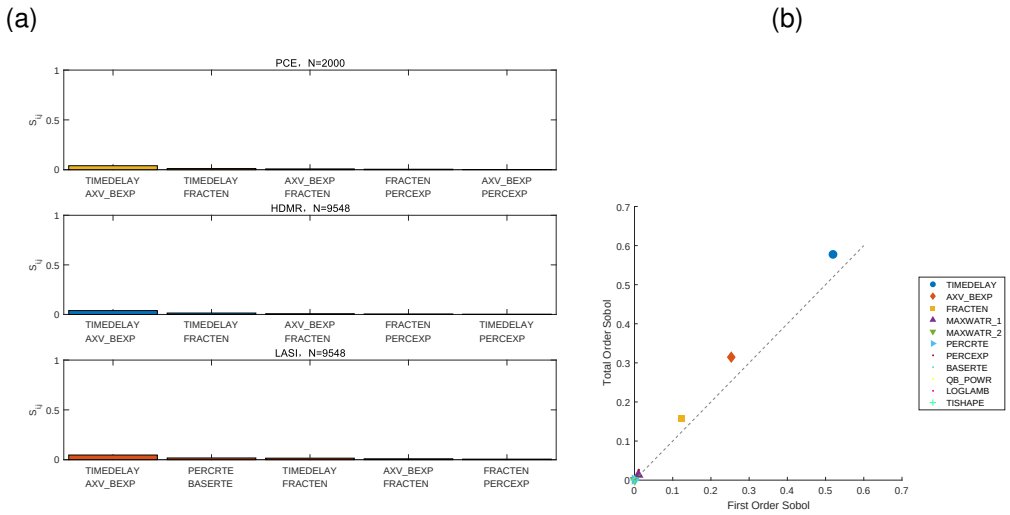


Figure 3.8: (a) Interaction quantification using the five highest estimates of second order Sobol' indices $S_{i,j}$ for the FUSE-016 model. Calculated using PCE, HDMR and LASI. All methods register the value of $S_{i,j}$ for i=TIMEDELAY and j=AXV_BEXP as highest second order index. However, the values of all second order Sobol' indices are small (the indices may run on a scale between 0 and 1), confirming that interactions have a limited effect on RMSE in this case. (b) First-order (S_i) plotted against the corresponding total-order (ST_i) Sobol' indices for the 11 FUSE-016 parameters. Estimation is performed using the PCE subroutine in UQ_Lab from the available uncertainty quantification sample. Values on the line indicate no parameter interaction.

The three most important parameters, TIMEDELAY, AXV_BEXP and FRACTEN, are involved in the most relevant second order interactions. In particular, the interaction between TIMEDELAY and AXV_BEXP is identified as the strongest second order interaction. However, we need to observe that the values of the second order Sobol' indices are all small. The last two panels of Figure 3.8 (a) (estimated via HDMR and LASI subroutines) report that all estimates of $S_{i,j}$ are less than 0.05. This confirms the observation stated at the beginning of this section, that interactions have limited influence in the determination of the RMSE for the configuration of interest.

To investigate further and interpret these results in terms of identifiability, we report results for the total order indices. In fact, according to Ratto et al. (2007), Saltelli et al. (2008) and others, parameters associated with a small main effect but having a high total effect evidence a lack of identifiability. To offer an intuitive visual interpretation, we refer to Figure 3.8 (b). Here, the value of first order Sobol' index for each model input is represented on the horizontal axis, while the vertical axis presents the value of total order Sobol' indices. Each dot in the

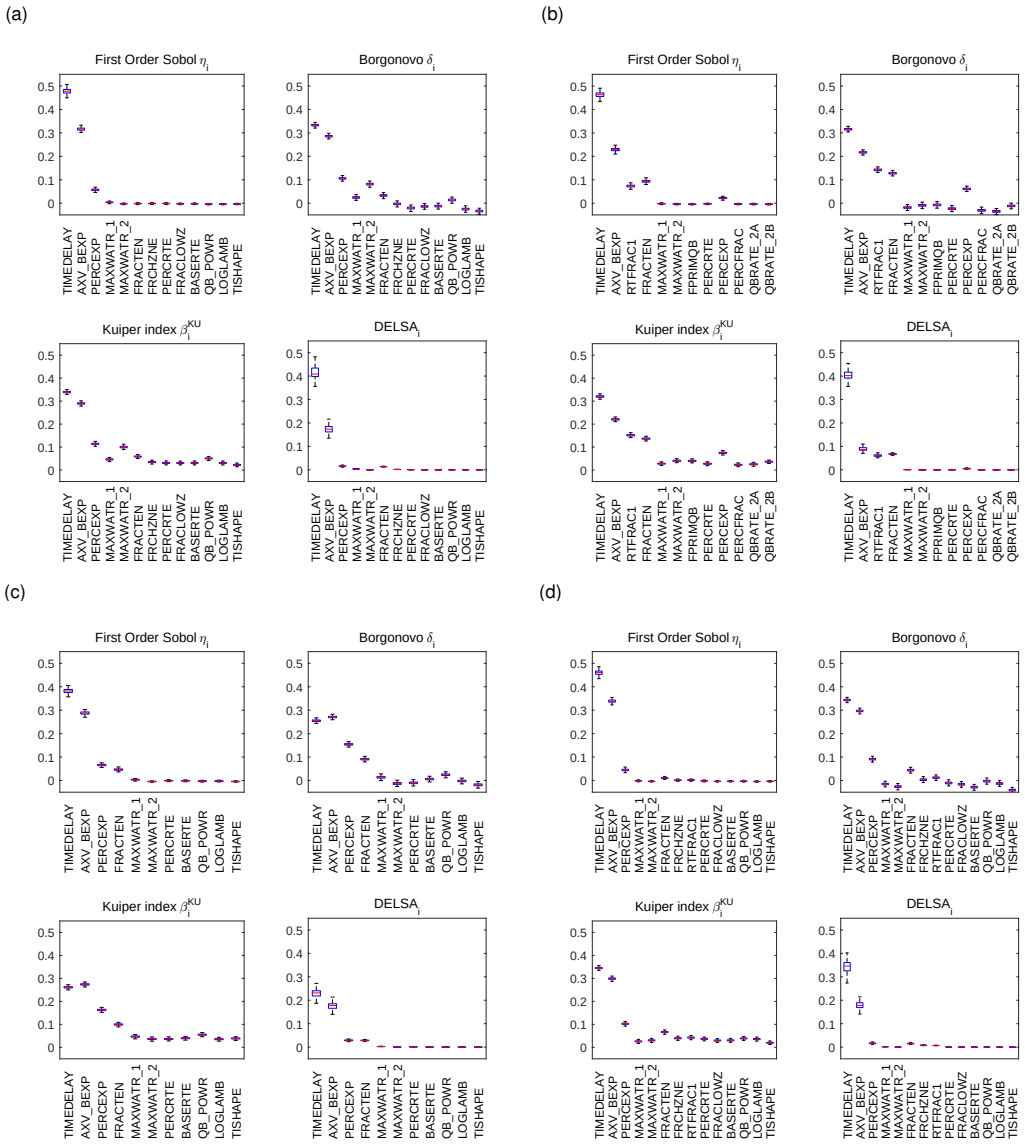


Figure 3.9: Factor prioritisation using the four sensitivity methods in Table 1 for alternative hydrologic configurations: (a) FUSE-014, (b) FUSE-160, (c) FUSE-072, and (d) FUSE-170.

graph corresponds to one parameter. Less identifiable parameters would lie further toward the top-left conner. For the FUSE-016 configuration, all parameters lie close to the 45-degree line, confirming the limited contribution of interaction effects. Thus, we register very limited identifiability issues for this configuration.

3.4.5 Alternative model configurations results

While in the description of results we have focused on the FUSE-016 configuration, we performed similar sensitivity analyses for the FUSE-014, FUSE-072, FUSE-160 and FUSE-170 models described by Rakovec et al. (2014). These results are summarized in Figures 3.9, 3.10 and 3.11.

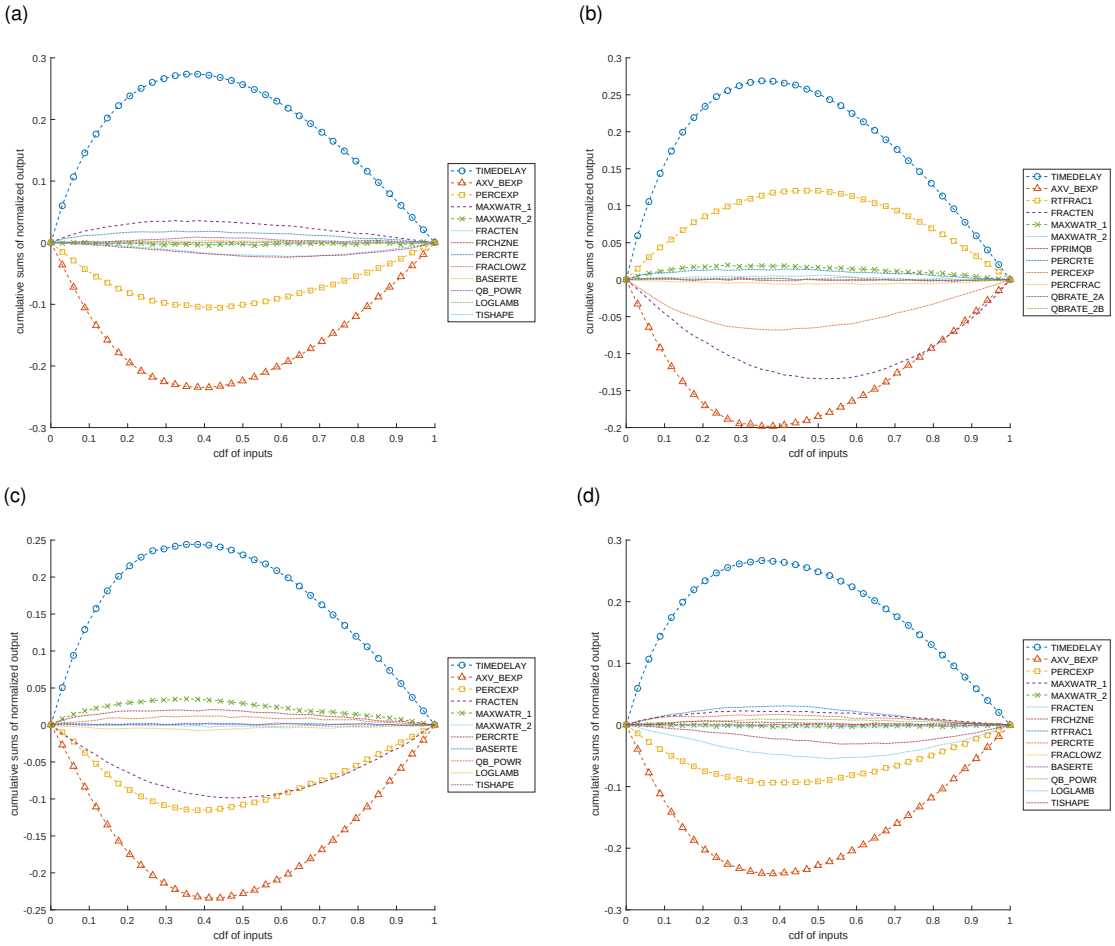


Figure 3.10: CUSUNORO curves for four alternative configurations: (a) FUSE-014, (b) FUSE-160, (c) FUSE-072, and (d) FUSE-170.

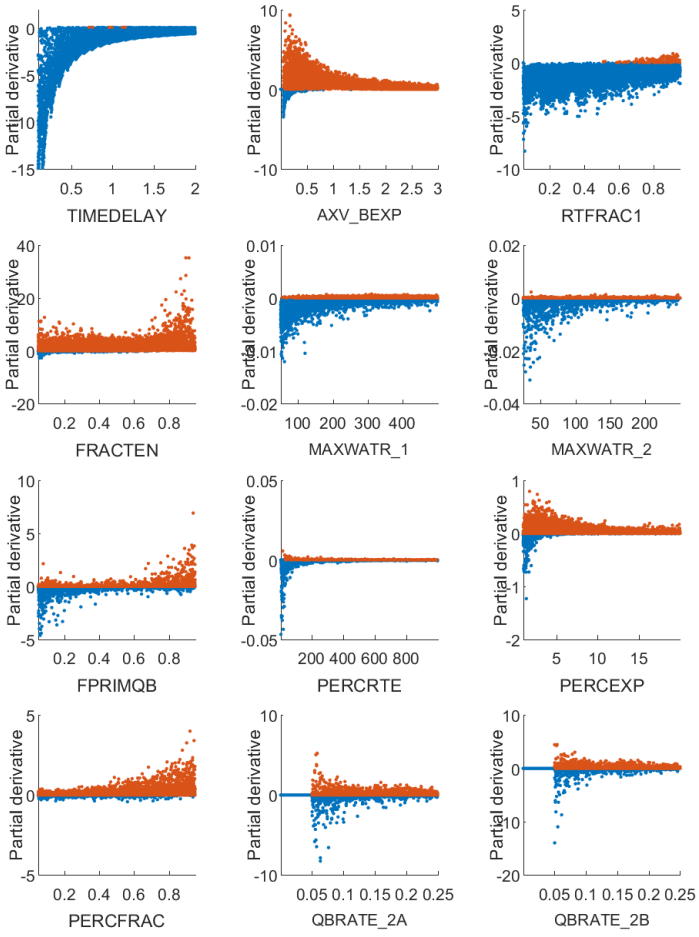


Figure 3.11: D-scatterplots for the parameters of the FUSE-160 model configuration. This graph provides complementary insights on Panel b in Figure S2. The FUSE-160 model RMSE is a non-monotonic function of most parameters. RTFRAC1 has a prevailing negative effect on RMSE.

We describe here the main differences in the sensitivity analysis results with respect to the FUSE-016 configuration, discussing them setting by setting. Regarding parameter prioritisation, Figure 3.9 shows that across all FUSE configurations the two most important parameters are TIMEDELAY and AXV_BEXP. The third most important parameter is PERCEXP for FUSE-014, FUSE-072, and FUSE-170 model. The parameter FRACTEN, the third most important for the FUSE-016 configuration, ranks fifth for FUSE-014 and fourth for FUSE-014,FUSE-072 and FUSE-170. For FUSE-160, FRACTEN is ranked third by first order Sobol' sensitivity measures and median DELSA indices, while it ranks fourth based on the moment-independent sensitivity indices δ and β^{Ku}. The parameter RTFRAC1 ranks third based on these sensitivity measures and fourth based on first order Sobol' sensitivity measures and median DELSA. This parameter is not present in the FUSE-016 configuration. For trend identification, we obtained the D-scatterplot, the COSI and CUSUNORO plots for all configurations. The CUSUNORO results are presented in the panels of Figure 3.10. The direction of change associated with the most important parameters remains the same across all the different configurations. Specifically on FUSE-160, the CUSUNORO curve of parameter RTFRAC1 shows a prevailing decreasing trend on the RMSE - see Figure 3.10, panel b. To test this indication further, consider the corresponding D-scatterplot (Figure 3.11). The first panel in Figure 3.11 shows that RTFRAC1 has a non monotonic effect, however, with a prevalence of negative partial derivatives. Positive values of the partial derivatives are only associated with high values of this parameter. Overall, the D-scatterplots of Figure 3.11 confirm the observations of the other configurations, though indicating that also in the FUSE-160 configuration the RMSE is not a monotonic function of the parameters.

Finally, for interaction quantification, we computed the second order and total order Sobol' indices measures using the PCE, HDMR and LASI subroutines. Results show a negligible effect of interactions in all configurations, with the sum of all second order indices never exceeding 6% of the total variance. This value indicates that identifiability issues are low and that the five model configurations are not over-parameterized, as suggested by Rosero et al. (2010).

3.5 Discussion

The investigation we have carried out leads to four takeaways for the analysis of hydrological models as well as for the practice of sensitivity analysis:

1. Given the huge number of available methods, the search for the appropriate sensitivity method needs to be made systematic. The most rigorous way to make the analysis systematic is through the formulation of a sensitivity analysis setting. A setting allows the analyst to transparently choose the method that answers the sensitivity analysis question at hand.

2. Employing a unique sensitivity method for performing a sensitivity analysis even within the same setting is suboptimal. In fact, each sensitivity analysis method has merits as well

as limitations. To illustrate, consider an analysis performed within a trend identification setting. Here, derivatives suggest the direction of change of the model output as a result of changes in the model parameters. However, they require scaling to suggest the importance of a parameter, especially if the parameters have different units — in that case, unscaled partial derivatives are not even comparable. An analyst willing to identify the most important parameters needs to consider sensitivity measures in a factor prioritisation setting. Here, a desirable property is that nullity of the sensitivity measure should imply independence. First order variance-based sensitivity measures, while appropriate, do not possess this property.

Thus, complementing the analysis with the calculation of a moment-independent measure increases the robustness of the inference. If the ranking obtained from variance-based methods is confirmed by the ranking implied by a moment-independent method, we gain additional confidence about what parameter is important, without the need of additional model runs.

3. Alternative methods may be applied under different circumstances. Consider again a trend identification setting. Partial derivatives are available through the DELSA method. However, in case derivatives are not available, one can use a CUSUNORO plot together with the plot of the first order effects of the functional ANOVA expansion to obtain insights about trend.

4. In the literature, the calculation of each global sensitivity measure requires a specific design. However, we can now estimate efficiently several sensitivity measures simultaneously from the same model output sample. This allows to exploit the synergies and complementary insights that different sensitivity measures make available to the modeler.

We believe that our study contributes to building a comprehensive sensitivity analysis framework which enables a thorough characterization of the most relevant sensitivity-related properties of model responses, as recently advocated by Razavi and Gupta (2015). Our approach can be extended to more complex environmental models currently being developed in the hydrological community, such as the next generation modeling framework SUMMA (Clark et al. (2015) and Clark et al. (2015)). SUMMA considers water and energy closure together and also allows to fully solve Richards' equation of the unsaturated flow. All these factors yield much higher degrees of input parameter uncertainty than in the presented study and the proposed framework can be directly employed for uncertainty quantification and sensitivity analysis of this new generation of hydrological models.
Finally, our analysis is applicable to the statistical diagnosis of models used in the broader environmental and climate literature. For example there is a growing interest in the sensitivity analysis of integrated assessment models for climate change. Works such as Confalonieri et al. (2010); Anderson et al. (2014); Butler et al. (2014a,b); Gao et al. (2016); Marangoni et al. (2017); Paleari and Confalonieri (2016) show the growing trend of sensitivity analysis investigations in the climate and environmental modelling arena, where the need for keeping computational burden under control is also felt. In this respect, the ensemble approach developed here might result

in supporting investigators in such sectors as well.

Chapter 4

Bayesian estimation of probabilistic sensitivity measures

In the presence of model uncertainty, analysts employ probabilistic sensitivity methods to identify the key-drivers of change in the quantities of interest. Simulation complexity, large dimensionality and long running times may force analysts to make statistical inference at small sample sizes. Methods designed to estimate probabilistic sensitivity measures at relatively low computational costs are attracting increasing interest. In Chapter 4, we propose a fully Bayesian approach to the estimation of probabilistic sensitivity measures based on a one-sample design. We discuss, first, new estimators based on placing piecewise constant priors on the conditional distributions of the output given each input, by partitioning the input space. We then present two alternatives, based on Bayesian non-parametric density estimation, which bypass the need for predefined partitions. In all cases, the Bayesian paradigm allows the quantification of uncertainty in the estimation process through the posterior distribution over the sensitivity measures, without requiring additional simulator evaluations. The performance of the proposed methods is compared to that of traditional point estimators in a series of numerical experiments comprising synthetic but challenging simulators, as well as a realistic application.
This chapter contains joint work with Isadora Antoniano-Villalobos and Emanuele Borgonovo, and is in preparation for submission.

4.1 Motivation

The use of computer simulations is becoming increasingly important in broad areas of science (Lin et al., 2010; Wong et al., 2017). High-fidelity mathematical models allow analysts to perform virtual (or *in silico*) experiments on complex natural or societal phenomena of interest (see Smith et al. (2009) among others). Predictions are often used to support policy-making. However, the level of sophistication of the models is often too high for analytical solutions to be available. In these cases, the only way to obtain a quantitative solution may be to encode complex mathematical equations in a computer software; so that the input-output mapping

remains a black-box to the analyst. It then becomes important to carefully design and execute the computer experiment. The design and analysis of computer experiments (DACE) has entered the statistical literature with the seminal work of Sacks et al. (1989) (see also the monographs of Santner et al. (2003); Kleijnen (2008)). Since then, researchers have studied the creation of space-filling designs (He, 2017), the calibration of computer codes with real data (Tuo and Wu, 2015), their emulation (Conti et al., 2009), the quantification of uncertainty in their output (Oakley and O'Hagan, 2002; Ghanem et al., 2016) and their sensitivity analysis (Oakley and O'Hagan, 2004; Borgonovo et al., 2014). These areas are intertwined. A given design may allow, for instance, not only an uncertainty quantification, but also the creation of an emulator and the analysis of sensitivity.

Probabilistic (or global) sensitivity measures are an indispensable complement of uncertainty quantification, as they highlight which areas should be given priority when planning data collection or further modelling efforts. International agencies such as the US Environmental Protection Agency (US EPA, 2009) or the British National Institute for Health Care Excellence (NICE, 2013) and the European Commission (2009) have issued guidelines recommending the use of probabilistic sensitivity analysis methods as the gold standard for ensuring reliability and transparency when using the output of a computer code for decision-making under uncertainty. Over the years, several probabilistic sensitivity measures have been proposed. Different measures enjoy alternative properties making them preferable in different contexts and for different purposes. We recall regression-based (Helton and Sallaberry, 2009), variance-based (Saltelli and Tarantola, 2002) and moment-independent measures (Borgonovo et al., 2014), all of which offer alternative ways to quantify the degree of statistical dependence between the simulator inputs and the output. A transversal issue in realistic applications is that analytical expressions of these measures are unavailable and analysts must resort to estimation. This, however, is a challenging task, especially for simulators with a high number of inputs (the *curse of dimensionality*) or with long running times (high *computational cost*).

Recent results (e.g Strong et al., 2012; Strong and Oakley, 2013) evidence the *one-sample* (or *given-data*) approach as an attractive design, which allows analysts to estimate global sensitivity measures from a single *probabilistic sensitivity analysis sample*, i.e., a sample generated for propagating uncertainty in the simulator. Thus, a one-sample approach has a nominal cost equal to the sample size and independent of the number of inputs, a feature that potentially reduces the impact of the curse of dimensionality. Related works such as Strong et al. (2012); Strong and Oakley (2013); Plischke et al. (2013); Strong et al. (2014, 2015) and Borgonovo et al. (2016) provide advances on theoretical and numerical aspects of the methodology, while at the same time, evidencing some open research questions. One-sample estimation procedures are closely related to scatter-plot smoothing, where partitioning of the covariate space plays a central role (Hastie and Tibshirani, 1990). Strong and Oakley (2013) show that the choice of partition size affects estimation, especially when the sample size is small (see Figure 1 of Strong and Oakley, 2013, p. 759). In the literature, some heuristics for determining a partition selection strategy which is optimal in some sense have been proposed, but finding a universally valid heuristic seems out of reach (see Section 4.3.2 for numerical experiments illustrating this

issue). Moreover, the literature is concerned with point estimators and uncertainty regarding the estimated value of a sensitivity measure is not an intrinsic part of the analysis. Because most one-sample estimators are consistent (in the frequentist sense), an accurate estimation of the error is often overlooked. However, especially at small sample sizes, it is essential for transparency that interval estimates become part of result communication (see Janon et al. (2014) among others).

We propose to enrich the one-sample design through the use of Bayesian non-parametric (BNP) methods, aiming to reduce and even eliminate the partition selection problem, while making uncertainty in the estimates a natural ingredient. First, we extend the partition approach, using Bayesian non-parametric models to augment the output sample within each partition set, by adequately generating additional synthetic data according to two alternative schemes. The Bayesian intuition supporting these designs can be interpreted as setting a prior on the conditional distribution of the output, given that the input realization falls within a given set of the partition. We build estimators based on this intuition for variance-based, density (pdf)-based and cumulative distribution function (cdf)-based global sensitivity measures. We compare the results with given-data estimators currently in use at low sample sizes, through numerical experiments. The results show that our estimators recover the correct ranking of the inputs, while providing an appropriate quantification of the estimation uncertainty. However, the results may be strongly influenced by the partition choice. Therefore, we investigate two additional classes of estimators based on Bayesian non-parametric joint and conditional density estimation methods. These estimators eliminate the partition selection problem and, at the same time, enable error quantification. Finally, we discuss the application of all the new estimators to the global sensitivity analysis of the benchmark computer code known as LevelE (Saltelli and Tarantola, 2002). Results show that the estimators correctly identify the key drivers of uncertainty at sample sizes lower than the ones used in previous literature. Additionally, the analyst obtains a quantification of the uncertainty in the estimates in the form of a posterior distribution which can be used to determine whether the available sample is large enough to make robust conclusions about the simulator input ranking.

The reminder of this chapter is organized as follows. We begin in Section 4.2 by introducing the framework of global sensitivity analysis and the one-sample estimation approach for probabilistic sensitivity measures. Section 4.3 combines Bayesian non-parametric methods and the one-sample approach to create two new partition-dependent estimators. Section 4.4 derives two classes of Bayesian partition-free estimators by adopting Bayesian a non-parametric density estimation approach. Section 4.6 presents numerical results for the LevelE code. Section 4.7 offers discussion and conclusions.

Table 4.1: Some probabilistic sensitivity measures

Measure	$\zeta(\mathbb{P}_Y, \mathbb{P}_{Y\|X^i})$	ξ_i
η_i	$(\mathbb{E}[Y\|X^i] - \mathbb{E}[Y])^2/\mathbb{V}[Y]$	$\mathbb{V}[\mathbb{E}(Y\|X^i)]/\mathbb{V}[Y]$
δ_i	$\frac{1}{2}\int_{\mathcal{Y}} \lvert f_{Y\|X^i}(y\|X^i) - f_Y(y)\rvert \mathrm{d}y$	$\frac{1}{2}\mathbb{E}\left[\int_{\mathcal{Y}} \lvert f_{Y\|X^i}(y\|X^i) - f_Y(y)\rvert \mathrm{d}y\right]$
β_i	$\sup_{y\in\mathcal{Y}} \left\lvert F_{Y\|X^i}(y\|X^i) - F_Y(y)\right\rvert$	$\mathbb{E}\left[\sup_{y\in\mathcal{Y}} \left\lvert F_{Y\|X^i}(y\|X^i) - F_Y(y)\right\rvert\right]$

4.2 Probabilistic sensitivity analysis of computer experiments

The notation for illustration in this chapter is consistent with that used in Chapter 2. To clarify, we recall the following. Formally, the sensitivity analysis framework considers a multivariate mapping $g : \mathcal{X} \mapsto \mathcal{Y}$ with input space $\mathcal{X} \subseteq \mathbb{R}^k$ and output space $\mathcal{Y} \subseteq \mathbb{R}^d$, denoted as $\mathbf{y} = g(\mathbf{x}) + \epsilon(\mathbf{x})$ in its more general form. In the DACE set-up, g represents a computer code which processes a set $\mathbf{x}$ of inputs, resulting in a set $\mathbf{y}$ of outputs of interest. The term $\epsilon(\mathbf{x})$ represents a zero-mean error term, which is present when the simulator response is stochastic. For simplicity, we focus on deterministic univariate responses, with $\epsilon(\mathbf{x}) \equiv 0$ and $d = 1$. In probabilistic sensitivity analysis, we denote the input probability space by $(\mathcal{X}, \mathcal{B}(\mathcal{X}), \mathbb{P}_{\mathbf{X}})$, where $\mathbb{P}_{\mathbf{X}}$ represents the joint probability measure of $\mathbf{X} = (X^1, \ldots, X^k)$, assumed known [1]. Similarly, $(\mathcal{Y}, \mathcal{B}(\mathcal{Y}), \mathbb{P}_Y)$ denotes the output probability space, where $\mathbb{P}_Y$ represents the distribution of Y induced by $\mathbb{P}_{\mathbf{X}}$ through g.

It has been recently shown that several probabilistic sensitivity measures frequently used in practice can be expressed as expectations of measures of discrepancy between $\mathbb{P}_Y$ and $\mathbb{P}_{Y|X^i}$. In particular, we focus on probabilistic sensitivity measures of the form:

$$\xi_i := \mathbb{E}[\zeta(\mathbb{P}_Y, \mathbb{P}_{Y|X^i})] \tag{4.1}$$

where the expectation is calculated with respect to the marginal distribution of X^i and ζ is a pre-metric on the space of probability measures over $\mathcal{Y}$, and ξ_i is called the *probabilistic sensitivity measure* of X^i with *inner operator* ζ (Borgonovo et al., 2014) [2].

Table 4.1 reports three probabilistic sensitivity measures encompassed by this construction, namely, the *variance-based* sensitivity measure (η_i), the *density-based* δ-importance measure (δ_i) and the *cdf-based* β-importance measure (β_i) (Pearson, 1905; Saltelli and Tarantola, 2002; Oakley and O'Hagan, 2004) [3]. For more details of probabilistic sensitivity measures, we refer the reader to Sections 2.2.3 and 2.2.4 of Chapter 2.

As mentioned in the Section 4.1, analytical expressions for these and other popular sensitivity measures are not available in most realistic applications, and their estimation is a prolific subject of research. We now discuss some relevant aspects of numerical estimation in the next

[1] For illustration, we use superscripts to indicate the input indices.

[2] Also see Section 2.2.3 of Chapter 2.

[3] The cdf-based importance measure is denoted as β_i^{KS} in Section 2.2.3 of Chapter 2. For ease of notation, we drop the superscript KS from β_i^{KS} in this chapter.

section.

4.2.1 Discussion on one-sample estimation

The estimation of global sensitivity measures is a challenging task and the availability of efficient designs is crucial in realistic applications. The number of simulator evaluations necessary to estimate sensitivity measures encompassed by Eq. (4.1) for a simulator with k simulator inputs, using a brute-force approach, would be of the order of $C = kn^2$ simulator runs, where n denotes the sample size required for Monte Carlo uncertainty quantification. The design becomes rapidly infeasible. For instance, if $k = 20$ and $n = 1,000$, the $C = 20,000,000$ simulator runs would require a prohibitive computational effort for most complex computer codes used in practice. However, notable advances in the literature have contributed in abating this computational burden, see Tissot and Prieur (2015); Janon et al. (2014) for reviews. Saltelli (2002a), for instance, achieved the estimation of variance-based sensitivity measures at a cost of $C = n(k+2)$ simulator runs, while the $FAST$ method of Saltelli et al. (1999) achieves a cost of order $C = nk$.

Recently, efforts have been made towards an estimation cost independent of the number of simulator inputs, k. Strong and Oakley (2013), Strong et al. (2014) and Strong et al. (2015) show that value-of-information measures can be estimated from a single probabilistic sensitivity analysis sample, $\{(\mathbf{x}_j, y_j) : j = 1, \dots n\}$, i.e., from the Monte Carlo sample generated for uncertainty quantification, thus lowering the computational cost to $C = n$ simulator runs. Roehlig et al. (2009) and Strong et al. (2012) obtain similar results for first-order variance-based sensitivity measures and Plischke et al. (2013) extend the intuition to density-based measures. The approaches proposed in these works receive the common name of *one-sample* or *given-data* estimation methods.

One-sample methods can be seen as generalizations of the intuition developed for estimating the correlation ratio (Pearson, 1905). If X^i is a discrete random variable then, an input-output sample of (sufficiently large) size n contains repeated observations of $Y = g(X^i, X^{-i})$, for each fixed value $X^i = x^i$, while the other factors, $X^{-i} = (X^1, \dots, X^{i-1}, X^{i+1}, \dots, X^k)$, remain random. This allows the estimation of $\zeta(\mathbb{P}_Y, \mathbb{P}_{Y|X^i=x^i})$ directly from a sample of size n. For a continuous X^i, a similar result may be achieved by partitioning the support $\mathcal{X}^i$ of X^i into M bins $\{\mathcal{X}^i_m\}_{m=1}^M$. The point condition $(X^i = x^i)$ is then replaced by the bin condition $(X^i \in \mathcal{X}^i_m)$. Then, for any sensitivity measure encompassed by Eq. (4.1), a one-sample estimator is given by (Borgonovo et al., 2016):

$$\widehat{\xi_i} = \sum_{m=1}^{M} \mathbb{P}_{X^i}(\mathcal{X}^i_m)\, \hat{\zeta}^i_m, \tag{4.2}$$

where $\hat{\zeta}^i_m$ may be any estimator of $\zeta(\mathbb{P}_Y, \mathbb{P}_{Y|X^i \in \mathcal{X}^i_m})$. Note that by using equiprobable partition sets, $\mathbb{P}_{X^i}(\mathcal{X}^i_m)$ should reduce to $1/M$. In practice, this partition probability is estimated by the sample proportion, n^i_m/n, where n^i_m denotes the number of realizations for which the i-th input falls within the m-th partition set of its support. Borgonovo et al. (2016, Theorem 2) show that, under mild conditions on the inner operator ζ, a consistent version of the estimator in Eq. (4.2)

can be obtained, if the size M of the partition is chosen as a monotonically increasing function of the sample size n, such that $\lim_{n\to\infty} \frac{n}{M(n)} = \infty$.

Indeed, the most popular one-sample estimator of η_i relies on a plug-in estimator of the inner statistic, based on the output sample mean and variance, $\bar{y}$ and s_y^2 respectively, to estimate the marginal mean and variance of Y. The within cluster sample mean $\bar{y}_m^i = \frac{1}{n_m^i}\sum_{y\in\mathbf{y}_m^i} y$ with $\mathbf{y}_m^i = \{y_j : x_j^i \in \mathcal{X}_m^i, j = 1, 2, ..., n\}$ is used to estimate the conditional mean of $Y|X^i \in \mathcal{X}_m^i$. The final expression (see e.g. Strong et al., 2012) takes the form of Eq. (4.2) with:

$$\widehat{\eta}_i^\star = \sum_{m=1}^{M} \frac{n_m^i}{n} \frac{(\bar{y}_m^i - \bar{y})^2}{s_y^2}. \tag{4.3}$$

The one-sample estimator for the $\delta-$importance introduced by Plischke et al. (2013) can be written as:

$$\widehat{\delta}_i^\star = \sum_{m=1}^{M} \frac{n_m^i}{n} \int_{\mathcal{Y}} |\hat{f}_Y^\star(y) - \hat{f}_m^i(y)| \mathrm{d}y, \tag{4.4}$$

where $\hat{f}_Y^\star$ and $\hat{f}_m^i$ denote kernel-smoothed histograms of the full output vector $\mathbf{y} = (y_1, \dots y_n)$ and the within cluster output vector $\mathbf{y}_m^i$, respectively. The authors propose a quadrature method for the numerical integration required by the L^1-distance in the inner operator, but other solutions could be used, producing similar estimators. Because estimates of this type rely on the approximation or estimation of probability density functions, we refer to them as *pdf-based estimators*.

Plischke and Borgonovo (2017) observe that the kernel-smoothing methods commonly involved in the calculation of pdf-based estimators may induce bias, even at high sample sizes, for simulators with a sparse output. Therefore, they introduce alternative *cdf-based estimators* which rely on the properties of empirical cumulative distribution functions.

Scheffé's theorem allows one to write the L^1-distance between two probability density functions in terms of the associated probability functions, as $\int_{\mathcal{Y}} |f_1(y) - f_2(y)| \mathrm{d}y = 2(\mathbb{P}_1(Y \in B) - \mathbb{P}_2(Y \in B))$, where B is the set of values for which $f_1(y) > f_2(y)$. Since B can be written as a union of intervals $(a(t), b(t))_{t=1}^T$, these probabilities can be calculated from the corresponding cumulative distribution functions. Thus, a cdf-based estimator of δ_i can be obtained as:

$$\widehat{\delta}_i^\diamond = \sum_{m=1}^{M} \frac{n_m^i}{n} \sum_{t=1}^{T_m^i} \left(\hat{F}_m^i(\hat{b}_m^i(t)) - \hat{F}_m^i(\hat{a}_m^i(t))\right) - \left(\hat{F}_Y(\hat{b}_m^i(t)) - \hat{F}_Y(\hat{a}_m^i(t))\right). \tag{4.5}$$

For further details on the estimation of the intervals $(\hat{a}_m^i(t), \hat{b}_m^i(t))$, we refer to Plischke and Borgonovo (2017).

Since β_i is itself a cdf-based sensitivity measure, the definition of a one-sample cdf-based estimator is straightforward:

$$\widehat{\beta}_i^\diamond = \sum_{m=1}^{M} \frac{n_m^i}{n} \max_{j\in\{1,\dots,n\}} \left|\hat{F}_Y(y_j) - \hat{F}_m^i(y_j)\right|, \tag{4.6}$$

where $\hat{F}_Y$, and $\hat{F}_m^i$ are the empirical cdf's of $\mathbf{y}$ and $\mathbf{y}_m^i$, respectively, i.e.:

$$\hat{F}_Y(y) = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}_{(-\infty,y_j]}(y); \qquad \hat{F}_m^i(y) = \frac{1}{n_m^i}\sum_{y_j \in \mathbf{y}_m^i} \mathbb{1}_{(-\infty,y_j]}(y), \tag{4.7}$$

and $\mathbb{1}_A(y)$ denotes the indicator function, taking the value 1 if $y \in A$ and 0 otherwise.

Recalling that the expected value of a random variable Y can be calculated as the integral of its survival function, $\mathbb{E}[Y] = \int_{\mathcal{Y}}(1 - F_Y(y))\mathrm{d}y$, a cdf-based one-sample estimator of the variance-based sensitivity measure, η_i is given by:

$$\widehat{\eta}_i^{\diamondsuit} = \sum_{m=1}^{M} \frac{n_m^i}{n} \frac{\left(\int_{\mathcal{Y}} \hat{F}_m^i(y) - \hat{F}_Y(y)\mathrm{d}y\right)^2}{\widehat{\sigma}_Y^2}. \tag{4.8}$$

Notice that, since the empirical distribution functions are piece-wise constant, the integral in the above expression reduces to a sum. Plischke and Borgonovo (2017) propose an efficient way to calculate this integral.

Most of the estimators found in the literature, including those mentioned above, are constructed either as deterministic approximations or as (frequentist) point estimators. Therefore, quantification of the estimation error (or interval estimation) requires additional manipulation. Finding asymptotic distributions of the estimators in order to provide approximate confidence intervals is not straight forward, except for the variance-based estimator $\eta_i^\star$, and even in this case, they are accurate only for high sample sizes. For instance, Gamboa et al. (2016); Janon et al. (2014); Tissot and Prieur (2015) show that variance-based estimators, calculated with a pick-and-freeze design or a replicated Latin hypercube, are asymptotically normal, but similar results are not available for other sensitivity measures. An alternative for non-deterministic sampling methods, is to replicate the estimation procedure in order to obtain a sample of estimates and corresponding sample-based confidence intervals. This, however requires a number $C > n$ of simulator evaluations and, as mentioned in the introduction, the computational cost of such an effort could be prohibitive for time-demanding realistic applications. The idea of replicates also excludes the use of quasi-random generators to create the probabilistic sensitivity analysis samples, as they are deterministic in nature. As a further alternative, bootstrap confidence intervals have been proposed in the literature in order to avoid the need for additional simulator runs (Plischke et al., 2013; Janon et al., 2014), but these are not an integral part of the estimation process.

A second issue to consider when using partition-based one-sample methods is the sample size bias induced by the partition. Quantities related to the marginal distribution of Y are estimated using the full sample size n, but those related to the within bin distribution of $Y|X^i \in \mathcal{X}_m^i$ are estimated using a smaller sample size $n_m^i \approx n/M$. While a sample size correction is implicit in the estimation of variances (see Eq. 4.3), the same is not true for the pdf and cdf estimates of equations (4.4) to (4.5). In other words, there is a different granularity when estimating the conditional and the unconditional distributions. In Section 4.3, we propose two partition-

dependent Bayesian estimators which mitigate the sample size bias, while providing a natural way to quantify the estimation error, allowing interval estimation.

Within the Bayesian paradigm, unknown objects are treated as random, and assigned a prior probability measure which reflects the analyst's uncertainty about their values. In this context, Oakley and O'Hagan (2004) treat the input-output mapping g as unknown (at least before evaluation). Thus, they define a semi-parametric regression model with a Gaussian process prior, which allows posterior inference on variance-based sensitivity measures. In fact, it is possible to calculate posterior means for the conditional and unconditional variance of Y and $Y|X^i$ respectively, either analytically or via numerical integration. The approach eliminates the need for a partition of the covariate space, thus solving the second issue mentioned above. However, the posterior distributions of the variance-based measures (e.g. η_i) are not available analytically, and finding a posteriori credibility intervals for estimation error quantification would be cumbersome and this aspect is not treated in the paper. Furthermore, it is not clear how to extend the results to the estimation of other (pdf or cdf-based) sensitivity measures. In Section 4.4, we present two alternative partition-free Bayesian models which allow interval estimation for these types of sensitivity measures as well. For illustrative purposes, we focus on estimation of the three measures in table 4.1.

4.3 Bayesian non-parametric partition-dependent estimation

We propose to quantify the uncertainty about fixed but unknown sensitivity measures, ξ_i, before (a priori) and after (a posteriori) the observation of a sample, $\{(\mathbf{x}_j, y_j) : j = 1, \dots n\}$, within the Bayesian paradigm. The ξ_i play the role of parameters of interest and they are linked to the data through functionals of the marginal and conditional distributions of Y and $Y|X^i$. In view of this, it seems sensible to induce a prior on ξ_i by assigning a prior to the family $\mathcal{P}_i = \{\mathbb{P}_{Y|X^i=x^i} : x^i \in \mathcal{X}^i\}$ of conditional probability measures. Notice that, since $\mathbb{P}_{X^i}$ is assumed known, the marginal distribution of Y, $\mathbb{P}_Y(y) = \int_{\mathcal{X}^i} \mathbb{P}_{Y|X^i=x^i}(y|x^i)\mathrm{d}\mathbb{P}_{X^i}(x^i)$, is fully determined by $\mathbb{P}_{Y|X^i}$, so no additional prior specification is required. For each i, $\mathcal{P}_i$ is a family of probability measures on $\mathcal{Y}$, indexed by $x^i \in \mathcal{X}^i$, so defining a prior probability on this space is, in principle, not a simple task. Furthermore, the relation between ξ_i and $\mathcal{P}_i$ is complex, making it difficult to conceive an adequate parametric prior. In other words, choosing a family of distributions characterized by a finite-dimensional parameter θ, to express an expert's uncertainty about ξ_i through some prior on θ would seem overly restrictive, if not unreasonable. It is known that an inadequate prior may lead to troublesome posterior (Freedman, 1965) and hinder the properties of the proposed estimators. A natural alternative is to use a Bayesian non-parametric prior in order to ensure enough flexibility to capture complex data structures. Bayesian non-parametric methods are not restricted to a finite number of parameters to represent a distribution. Generally speaking, they rely on measure-valued stochastic processes to define priors on the space of

probability measures of interest. The supports of such priors are wide, ideally covering the full range of all possible distributions, in our case, over $\mathcal{Y}$ (see e.g. Hjort et al., 2010, for an extensive discussion on bayesian non-parametric priors, their properties and their use).

Our first proposal can be interpreted as a Bayesian refinement of the cdf-based estimators introduced in the previous section and, as such, relies on a partition of the input space. We assume that the distribution of $Y|X^i = x^i$ is identical for every $x^i \in \mathcal{X}^i_m$, and denote it by $\mathbb{P}^i_m$. In practice, it is enough to assume that $\mathbb{P}_{Y|X^i=x^i}$ can be well approximated in this way. Prior uncertainty is expressed through a prior on the collection $\{\mathbb{P}^i_m\}_{m=1}^M$. For simplicity, we assume that such distributions are independent and identically distributed (i.i.d.), so the problem becomes that of finding a prior which assigns probability 1 to a large enough set of probability distributions supported on $\mathcal{Y}$. We focus our attention on the *Dirichlet Process* (DP), first introduced by Ferguson (1973) and widely studied in the BNP literature (see e.g. Hjort et al., 2010, Chapter 2, for a discussion on its properties). We therefore define, for each $i = 1, \dots k$ the following Bayesian non-parametric model:

$$Y|(\mathbb{P}^i_m, X^i \in \mathcal{X}^i_m) \sim \mathbb{P}^i_m; \qquad \mathbb{P}^i_m \overset{iid}{\sim} \mathcal{DP}(\alpha G), \tag{4.9}$$

where $\mathcal{DP}(\alpha G)$ denotes a Dirichlet process with base measure G and concentration parameter α. The Dirichlet process could be replaced by a more general stick-breaking process, achieving greater flexibility at a similar computational cost (see e.g. Ishwaran and James, 2001; Pitman and Yor, 1997; Lijoi et al., 2007). In this case, the algorithms and proposed estimators would maintain a similar structure so we focus on the Dirichlet process, without loss of generality, in order to use a notation more familiar to a wider audience. With regards to the hypothesis of independence between the $\mathbb{P}^i_m$, it could be removed through the application of recent developments in BNP methods (see Wood et al. (2011); Teh et al. (2006); Teh and Jordan (2010); Camerlenghi et al. (2017) and Camerlenghi et al. (2018)). This, however, would lead to a complication of the estimation algorithms which goes beyond the scope of this paper.

Note that this Bayesian model is coherent, in the sense that it induces a unique prior over the unconditional distribution of Y, whenever the partitions are equiprobable, that is when $\mathbb{P}(X^i \in \mathcal{X}^i_m) = \frac{1}{M}$ for all $i = 1, 2, ..., k$ and $m = 1, 2, ..., M$. In fact,

$$\mathbb{P}_Y(\cdot|\mathbb{P}^i_{1:M}) = \sum_{m=1}^{M} \mathbb{P}^i_m(\cdot)\mathbb{P}(X^i \in \mathcal{X}^i_m) = \frac{1}{M}\sum_{m=1}^{M} \mathbb{P}^i_m(\cdot).$$

Then, by marginalizing, we obtain

$$\mathbb{P}_Y(\cdot|\alpha G) = \frac{1}{M}\sum_{m=1}^{M} \int \mathbb{P}^i_m(\cdot)\mathrm{d}\mathcal{DP}(\mathbb{P}^i_m|\alpha G) = \int \mathbb{P}(\cdot)\mathrm{d}\mathcal{DP}(\mathbb{P}|\alpha G),$$

because $\int \mathbb{P}^i_m(\cdot)\mathrm{d}\mathcal{DP}(\mathbb{P}^i_m|\alpha G)$ does not depend on i or m. In other words, a priori, $\mathbb{P}_Y \sim \mathcal{DP}(\alpha G)$, so that the prior for the marginal simulator distribution is also a Dirichlet process. This statement alone, however, provides no information on the probabilistic dependence of Y

on X^i. Thus, it is not meaningful, by itself, for a sensitivity analysis.

The posterior for this model, given the simulator input-output realizations (*Data* for short), $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, can be written as follows:

$$Y|(X^i \in \mathcal{X}^i_m, \mathbb{P}^i_m) \sim \mathbb{P}^i_m; \qquad \mathbb{P}^i_m|Data \overset{ind}{\sim} \mathcal{DP}\left((\alpha + n^i_m)\widetilde{G}^i_m\right), \tag{4.10}$$

where

$$\widetilde{G}^i_m = \mathbb{E}[\mathbb{P}^i_m|Data] = \frac{\alpha}{\alpha + n^i_m} G + \frac{n^i_m}{\alpha + n^i_m} \sum_{y \in \mathbf{y}^i_m} \frac{1}{n^i_m} \delta^{Dirac}(y) \quad . \tag{4.11}$$

Note that the posterior of the marginal for Y can be obtained as:

$$\mathbb{P}_Y(\cdot|\alpha G, Data) = \frac{1}{M} \sum_{m=1}^{M} \int \mathbb{P}^i_m(\cdot) \mathrm{d}\mathcal{DP}(\mathbb{P}^i_m|(\alpha + n^i_m)\widetilde{G}^i_m), \tag{4.12}$$

which may depend both on i and m. However, the marginal coherence of the model still holds, at least asymptotically. Informally, for an equiprobable partition, $\mathbb{P}(X^i \in \mathcal{X}^i_m) = 1/M$, $n^i_m \simeq n/M$ when the sample size n is sufficiently large, so $\alpha/(\alpha + n^i_m) \simeq M\alpha/(M\alpha + n)$ and $n^i_m/(\alpha + n^i_m) \simeq n/(M\alpha + n)$. Furthermore, $\sum(1/n^i_m)\delta^{Dirac}(y) \simeq \sum(M/n)\delta^{Dirac}(y)$. Thus, asymptotically, $\mathbb{P}_Y(\cdot|\alpha G, Data)$ does not depend on m or i and $\mathbb{P}_Y(\cdot|\alpha G, Data) \sim \mathcal{DP}((\alpha + n)\widetilde{G})$, where

$$\widetilde{G} = \frac{\alpha}{\alpha + n} G + \frac{n}{\alpha + n} \widehat{\mathbb{P}}_n, \tag{4.13}$$

and $\widehat{\mathbb{P}}_n$ denotes the empirical distribution of Y based on the full set of observations, $(y_1, \ldots, y_n)$. Note that this is the usual posterior corresponding to the DP prior on $\mathbb{P}_Y$.

The sensitivity measures we aim to estimate are functionals of the conditional and marginal distributions. The posterior means in Eqs. (4.11) and (4.13), respectively, may be proposed as Bayesian estimators of such densities. Thus, a Bayesian point estimator of ξ_i may be given by:

$$\widetilde{\xi}_i = \sum_{m=1}^{M} \frac{n^i_m}{n} \zeta(\widetilde{G}, \widetilde{G}^i_m)$$

Unfortunately, the direct calculation of $\widetilde{\xi}_i$ is impractical. Moreover, our purpose is to provide interval estimation, so as to quantify the uncertainty associated to point estimates. A way out is to sample observations (i.e., predicted realizations of the output) from $\widetilde{G}$ and $\widetilde{G}^i_m$, in order to enrich the sample. More specifically, we have a vector $\mathbf{y}$ of n observations from the original simulator used to estimate $\mathbb{P}_Y$, but only n^i_m of these belong to $\mathbf{y}^i_m$ and are therefore used to estimate $\mathbb{P}^i_m$. Because $n^i_m < n$, the precision issue discussed in Section 4.2.1 emerges, causing a bias in the empirical estimation of ξ_i. By re-sampling from $\widetilde{G}$ and $\widetilde{G}^i_m$ we can enlarge both vectors, making them of the same size and, potentially, arbitrarily large. Our proposal here is simply to sample $n - n^i_m$ observations from $\widetilde{G}^i_m$, thus obtaining two vectors of size n. The intuition underlying this corresponds to the non-parametric *Bayesian bootstrap* (Bb) (Hjort, 1985, 1991). In our case, for each m a sample $\widetilde{\mathbf{y}}^i_m = \{\widetilde{y}^i_{n^i_m+1}, \ldots, \widetilde{y}^i_n\}$ of size $n - n^i_m$

is obtained from $\widetilde{G}_m^i$. A value of $\widehat{\xi}_i^{Bb,s}$ in Eq. (4.2) can be calculated through any of the methods discussed in Section 4.2.1, using $\mathbf{y}$ to estimate all quantities related to the marginal distribution of Y and the extended vector $\mathbf{y}_m^{Bb,i,s} = (\mathbf{y}_m^i, \widetilde{\mathbf{y}}_m^{i,s})$ to estimate all quantities related to the conditional distribution of $Y|X^i \in \mathcal{X}_m^i$. Informally, the weighted average over m can be seen as approximately simulated from the posterior distribution of ξ_i. By repeating this procedure S times, we obtain a Bb sample $\{\widehat{\xi}_i^{Bb,s} : s = 1, 2, ..., S\}$. We propose the Monte Carlo average:

$$\widehat{\xi}_i^{Bb} = \frac{1}{S}\sum_{s=1}^{S} \widehat{\xi}_i^{Bb,s}$$

as a point estimator of ξ_i. Approximate credibility intervals can be obtained from the empirical quantiles. Note that, because each $\widetilde{y}_j^i$ is simulated from a single distribution, $\widetilde{G}_m^i$, the sampling process can be done in parallel and the method is computationally fast. However, the uncertainty is underestimated because the additional variability captured by the posterior distribution of Eq. (4.10) is ignored.

A more accurate alternative is to sample $\widetilde{\mathbf{y}}_m^i$ jointly from the Dirichlet process posterior distribution (4.10), instead of sampling each $\widetilde{y}_j^i$ from the posterior mean. This can be done via the *Pólya Urn scheme* (Pu) of Blackwell and MacQueen (1973). Specifically, $\widetilde{\mathbf{y}}_m^i$ is generated as a realization of the Pólya sequence:

$$\widetilde{Y}_{j+1}^i | \left(\widetilde{y}_{n_m^i+1:j}^i, Data\right) \sim \frac{\alpha}{\alpha + j} G + \frac{j}{\alpha + j}\widehat{\mathbb{P}}_j \quad \forall\, j \geq n_m^i. \tag{4.14}$$

Once again, the extended samples $\mathbf{y}_m^{Pu,i,s} = (\mathbf{y}_m^i, \widetilde{\mathbf{y}}_m^{i,s})$ can be used to obtain a value $\widehat{\xi}_i^{Pu,s}$ by any available method to calculate the expression in Eq. (4.2). We use $\widehat{\xi}_i^{Pu}$ to denote the Monte Carlo average of a sample of size S generated in this way. Note that this is a point estimator with the same expectation as $\widehat{\xi}_i^{Bb}$. However, a greater variability which fully accounts for the uncertainty on $\mathbb{P}_m^i$ results in wider credibility intervals. The sampling procedure is now sequential for $s = 1, \ldots, S$, so the price for greater accuracy in uncertainty estimation is a slightly higher computational time.

The technical details for Bb and Pu estimators are presented in Section 4.5.1.

4.3.1 Simulation study

We illustrate the performance of the two classes of estimators proposed above, via two toy examples for which the sensitivity measures can be calculated analytically (see Table 4.2). For illustration, we summary the use of notation in Table 4.3. The first example is the 2-input simulator

$$Y = \frac{X^1}{X^1 + X^2}, \tag{4.15}$$

Table 4.2: Analytical values of η_i, δ_i and β_i for the two test simulators used in this section.

	2-input simulator		21-input simulator		
Sensitivity measure	X^1	X^2	$X^1 \dots X^7$	$X^8 \dots X^{14}$	$X^{15} \dots X^{21}$
η_i	0.496	0.496	0.109	0.027	0.007
δ_i	0.315	0.315	0.112	0.053	0.026
β_i	0.289	0.289	0.112	0.053	0.026

Table 4.3: Estimators used in numerical experiments

	One-sample		Partition-dependent Bayesian		Partition-independent Bayesian	
	pdf-based	cdf-based	Bayesian bootstrap	Pólya urn	joint	conditional
η	$\widehat{\eta}_i^{\star}$	$\widehat{\eta}_i^{\diamond}$	$\widehat{\eta}_i^{Bb}$	$\widehat{\eta}_i^{Pu}$	$\widehat{\eta}_i^{BNJ}$	$\widehat{\eta}_i^{BNC}$
δ	$\widehat{\delta}_i^{\star}$	$\widehat{\delta}_i^{\diamond}$	$\widehat{\delta}_i^{Bb}$	$\widehat{\delta}_i^{Pu}$	δ_i^{BNJ}	$\widehat{\delta}_i^{BNC}$
β	N/A	$\widehat{\beta}_i^{\diamond}$	$\widehat{\beta}_i^{Bb}$	$\widehat{\eta}_i^{Pu}$	$\widehat{\beta}_i^{BNJ}$	$\widehat{\beta}^{BNC}$

where X^1,$X^2 \overset{iid}{\sim}$ Gamma$(3,1)$, so that the output Y follows a Beta distribution. The second example is the 21-input simulator

$$Y = \sum_{i=1}^{21} a_i X^i, \tag{4.16}$$

where $X^i \overset{iid}{\sim}$ Normal$(1,1)$, with $a_1 = \dots = a_7 = -4$, $a_8 = \dots = a_{14} = 2$, and $a_{15} = \dots = a_{21} = 1$, so that Y is normally distributed.

We are interested in small sample sizes, which make the estimation of global sensitivity measures challenging. In particular, we consider $n = \{300, 600, 900\}$. The input data, $\mathbf{x}$, is generated via Quasi-Monte Carlo. For each n, alternative choices of the partition size, M, are explored. The mass parameter, α, for the DP prior is set equal to $0.1n/M$ throughout. The base measure, G, is chosen in correspondence with the support of Y: a Beta distribution for the first example and a Normal distribution for the second; the hyper-parameters are fixed through an empirical approach, based on the available sample $\mathbf{y}$. Note that this choice centres the prior distribution for $Y|X^i \in \mathcal{X}_m^i$ roughly around the marginal distribution of Y, thus favouring, a priori, independence between the Y and X^i, with a precision proportional to the number of observations in each partition set. In practical applications, prior information elicited from experts may be expressed through different choices of α and G.

We compare the Bayesian bootstrap and Pólya urn estimators to traditional point estimators for three global sensitivity measures. Results are reported in Figures 4.1 and 4.2: the first row corresponds to η_i, the second to δ_i and the third to β_i. Columns, from left to right, correspond to increasing sample sizes. Each graph is divided into three blocks displaying importance measures estimates based on alternative choices of M. The dotted lines display the analytical values.

We first consider the left-most panel of Figure 4.1(a). At $n = 300$ the estimates vary notably with the partition size: they are downward biased for $M = 3$ and upward biased for $M = 21$. Observe that at $M = 21$, we have $n_m^i \simeq 9$, a number too small to be reasonably chosen by the analyst. However, the bias is systematic, that is, it affects identically all estimates. Estimates are less affected by the partition choice as the sample size increases. Recall that in realistic

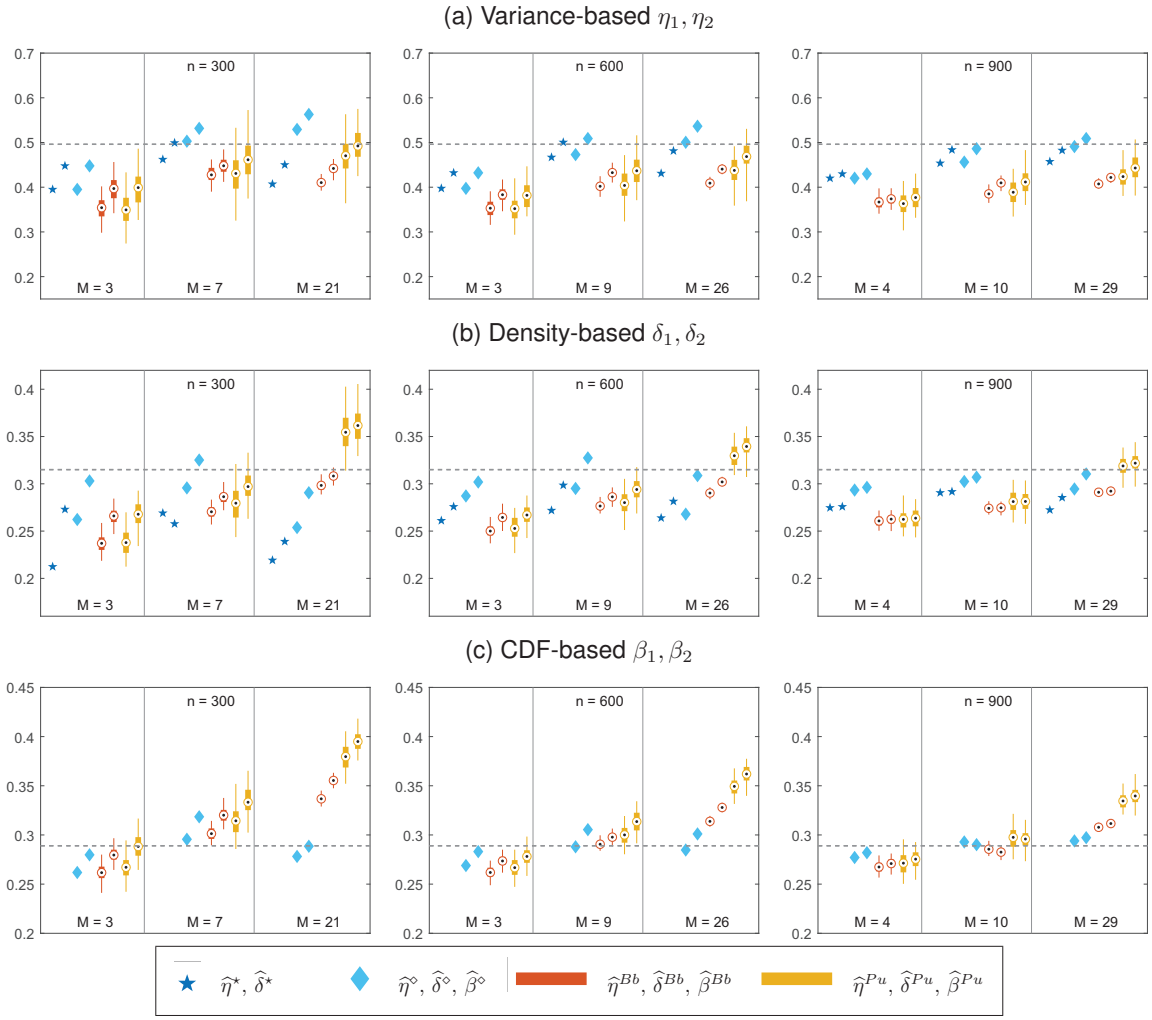


Figure 4.1: Results for the 2-input simulator in Eq. (4.15): comparison of sensitivity measures estimates using frequintist pdf/cdf-based estimators and partition-dependent Bayesian non-parametric estimators. Bayesian estimates include 95% credibility intervals.

applications, where an analyst would not know the true values of the sensitivity measures, the main interest is on the ordinal ranking of the inputs. In this example, X^1 and X^2 are equally important. However, looking at the point estimators $\widehat{\eta}_i^{\star}$ and $\widehat{\eta}_i^{\diamond}$ the analyst would rank X^2 as more important than X^1 for most combinations of n and M. The credibility intervals for $\widehat{\eta}_i^{\mathrm{Pu}}$ display a large overlapping that would prevent the analyst from ranking X^2 above X^1: there is too much uncertainty in the estimates to make such conclusion. Notice the underestimation of the uncertainty surrounding $\widehat{\eta}_i^{\mathrm{Bb}}$. Rows (b) and (c) of Figure 4.1 show a similar behaviour for the δ_i and β_i sensitivity measures.

Figure 4.2 shows the estimates for the 21−input simulator in (4.16). For a better display clarity, instead of reporting seven sensitivity measures per group, we show numerical values for a representative of each input group, namely X^3, X^{10} and X^{18}. The results in Figure 4.2 are in line with the ones for the 2−input simulator. Once again, one observes an upward bias in the

Figure 4.2: Results for the 21−input simulator in Eq. (4.16): comparison of sensitivity measures estimates using frequintist pdf/cdf-based estimators and partition-dependent Bayesian non-parametric estimators. Bayesian estimates include 95% credibility intervals.

estimates when M is high and a downward bias when M is low. For variance-based sensitivity measures (Figure 4.2 a) for all n and M considered we are able to correctly identify the group of inputs represented by X^3, corresponding to a higher absolute value of the coefficient in the simulator, as more important than X^{10} and X^{18} using the estimates $\widehat{\eta}_i^{\diamond}$, $\widehat{\eta}_i^{\mathrm{Bb}}$ and $\widehat{\eta}_i^{\mathrm{Pu}}$. However, $\widehat{\eta}_i^{\star}$ fails to indicate the correct ranking at $n = 300$, with M equal to 3 and 21, respectively. Regarding the sensitivity measures δ_i and β_i, in most cases the overlapping credibility intervals of Bayesian estimates would not allow us to deem X^{10} more relevant than X^{18}. Thus higher sample sizes would be needed for neatly ranking the second and third most important groups of simulator inputs.

Overall, Figures 4.1 and 4.2 suggest that the proposed estimators allow the identification of the most important inputs, even at small sample sizes, and, most relevantly, they provide a measure of the uncertainty in the assessment. However, the results also display a strong dependence on the partition size M. While *i)* as observed in Strong and Oakley (2013) (see their figure 1,

at p. 759), the importance of selecting an optimal partition size diminishes as the sample size increases and *ii)* a suboptimal partition selection has in most cases an identical impact on the sensitivity measures (i.e., the sensitivity measures of all inputs are simultaneously upward or downward biased), the analyst is still left with the question of what is the optimal partition size for a given sample. Unfortunately, there seems to be no universally optimal selection rule (see Section 4.3.2 for illustration).

4.3.2 Numerical experiments for the partition selection problem

We performed several thought experiments on test cases. The results show the difficulty, maybe impossibility, of finding a universally valid rule for linking the partition size M to the sample size n. We report some experiments results.

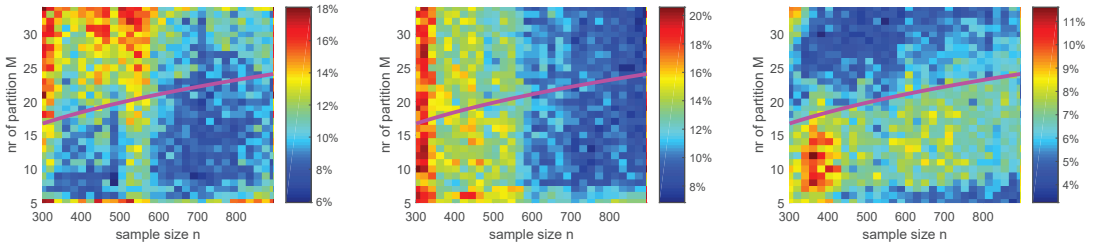
Assume the analyst wants to find an "optimal "(in some sense) partition refining strategy, i.e., a relationship that produces the partition size M that minimizes the estimation error at sample size n for the pdf-based point estimators $\widehat{\eta}_i^{\star}$, $\widehat{\delta}_i^{\star}$ and cdf-based point estimator $\widehat{\beta}_i^{\diamond}$ (Eqs. (4.3), (4.4) and (4.6)). We focus on one estimator type for simplicity and also because Borgonovo et al. (2016) propose a heuristic inspired by the rule of histogram partitioning of Freedman-Diaconis (Freedman and Diaconis, 1981), in which $M \sim \sqrt[3]{n}$.

To evaluate the estimators' performance at fixed values of M and n, we use the Root Mean Square Error (RMSE):

$$\mathrm{RMSE}_i(n) \approx \sqrt{\frac{\sum_{s=1}^{S}\left(\widehat{\xi}_i^s(n) - \xi_i\right)^2}{S}}$$

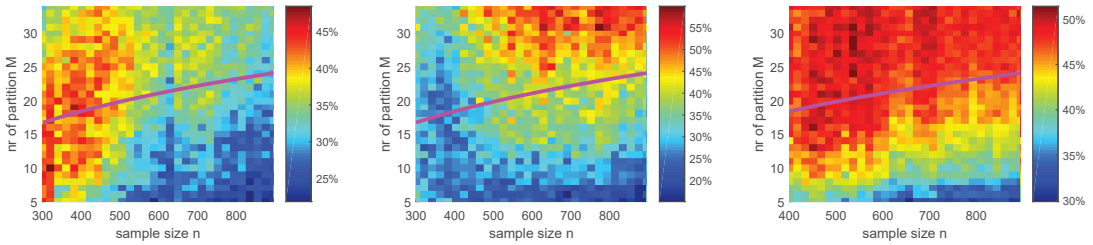
where S is the number of bootstrap replicates. $\widehat{\xi}_i^s$ is the s-th bootstrap replicate of ξ_i.

We estimate the sensitivity measures with sample sizes varying from 300 to 900, and partition sizes covering the natural numbers between 5 and 35. Then we calculate the RMSEs with $S = 100$ bootstrap replicates. Figures 4.3 and 4.4 present the heatplot of RMSEs in percentage ($\mathrm{RMSE}_i/\xi_i \cdot 100\%$). The horizontal axis indicates the sample size, and the vertical axis the partition size. The darker the color of a region in the plot, the lower the estimation error. For example, in Figure 4.3(a), dark (blue) refers to low RMSE (less than 10 percent), and light (red) to relative high RMSE (higher than 14 percent). The magenta line maps n into M using the previously mentioned heuristic function. Figure 4.3 shows that the proposed heuristic works well on the 2-input simulator (Eq. (4.15)), with the magenta line falling mainly into dark coloured regions. However, for the 21-input simulator (Eq. (4.16)) we would incur in high errors at small sample sizes. For instance consider graph a) in Figure 4.4. The graph reports the error in the estimates of δ_3 for the second model. The heuristic would propose values of M at about 20 for all values of n as optimal partition sizes. However, the partition size that minimizes the error is at about $M = 10$ or lower. The different behavior here could also be related to the differences in structure and dimensionality of the models. However, even for the same model, the heatplots differ significantly across the sensitivity measures. For the first simulator (Eq. (4.15)), the ideal partition size for the variance-based estimator is between 10 to 15 (Figure 4.3 (b)), while for



(a) Variance-based $\widehat{\eta}_1^\star$ (b) Density-based $\widehat{\delta}_1^\star$ (c) CDF-based $\widehat{\beta}_1^\diamond$

Figure 4.3: RMSE of sensitivity measures estimates for X^1 of the 2-input simulator in Eq. (4.15). Magenta lines correspond to $M = 2.5\sqrt[3]{n}$; $n \in [300, 900]$, $M \in [5, 34]$



(a) Variance-based $\widehat{\eta}_3^\star$ (b) Density-based $\widehat{\delta}_3^\star$ (c) CDF-based $\widehat{\beta}_3^\diamond$

Figure 4.4: RMSE of sensitivity measures estimates for X^3 of the 21-input simulator in Eq. (4.16). Magenta lines correspond to $M = 2.5\sqrt[3]{n}$; $n \in [300, 900]$, $M \in [5, 34]$

mutual information, if falls between 20 to 25 (Figure 4.3 (c)).

These results show that aiming at postulating a universally valid heuristic might be a cumbersome task. Clearly, the problem would be solved if partition-independent estimators were available. In Section 4.4, we study two proposals of Bayesian estimators that avoid the partition choice problem.

4.4 Bayesian non-parametric partition-free estimation

In this section, we propose two classes of Bayesian partition-free estimators. The first is based on the use of an infinite mixture model to estimate the joint density of Y and X^i. The second, uses a Bayesian non-parametric regression model to estimate the conditional density of Y given X^i.

4.4.1 Joint density-based estimation

The intuition is that all sensitivity measures under consideration can be recovered from the joint distribution of Y and X^i. Therefore, in order to do Bayesian inference on ξ_i it suffices to place a prior on the joint density $f_{X^i,Y}$. We propose to do so by means of a nonparametric mixture model (see, e.g. Ferguson (1983); Lo and Others (1984)). In other words, we consider $f_{X^i,Y}$ to

be defined as a mixture:

$$f_{X^i,Y}(\cdot,\cdot)|P = \int \mathcal{K}(\cdot,\cdot|\theta)\mathrm{d}P(\theta), \tag{4.17}$$

where $\mathcal{K}$ is a parametric bivariate density and the mixing measure P is a probability distribution over an appropriate space of parameters. The model is completed by assigning a non-parametric prior, Π, on P. Most common choices of Π assign probability one to discrete distributions of the form

$$P(\theta) = \sum_{\ell=1}^{\infty} w_\ell \, \delta^{Dirac}(\theta_\ell), \tag{4.18}$$

placing mass w_ℓ on locations (θ_ℓ). In the literature, particular attention has been paid to nonparametric priors admitting a stick-breaking construction (Pitman, 1996; Sethuraman, 1994) where the weights $\underline{w} = (w_1, w_2, ...)$ are defined as realization of random variables satisfying

$$W_1 = V_1, \quad W_\ell = V_\ell \prod_{\ell'=1}^{\ell-1}(1 - V_{\ell'}) \tag{4.19}$$

and independent of $\underline{\theta} = (\theta_1, \theta_2, ...) \overset{iid}{\sim} G$. Rich families of stick-breaking priors can be defined via different distributional assignments for the sequence $(V_1, V_2, \ldots)$ (see e.g. Favaro et al., 2012; Ishwaran and James, 2001). The main advantage over other types of construction is that the stick-breaking representation of the random weights allows for efficient simulation algorithms, specially in the context of nonparametric mixture models (Ishwaran and James, 2001; Papaspiliopoulos and Roberts, 2008; Kalli et al., 2011; Yau et al., 2011). However, the most popular stick-breaking prior remains the Dirichlet process, well known even outside the specialized community of Bayesian nonparametrics. For this reason, we will focus our analysis on DP mixtures, thus letting $P \sim \Pi = \mathcal{DP}(\alpha G)$. Additionally, for simplicity, we choose $\mathcal{K}$ to be a bivariate normal density, following the density estimation scheme of Escobar and West (1995). In this case, $\theta_\ell = (\mu_\ell, \Sigma_\ell)$ and, to simplify calculations, we select G as a conjugate Normal inverse-Wishart distribution. Thus, the the integral in (4.17) reduces to a sum and the joint density can be written as:

$$f_{X^i,Y}(\cdot,\cdot)|P = \sum_{\ell=1}^{\infty} w_\ell \cdot \mathcal{N}(\cdot,\cdot|\mu_\ell, \Sigma_\ell), \tag{4.20}$$

where the weights follow (4.19), with $V_i \overset{iid}{\sim} \mathrm{Beta}(1, \alpha)$.

Inference on this model is usually achieved via an MCMC scheme resulting in a sample from the posterior distribution of $f_{X^i,Y}$ given the *Data*. In the case of the DP-mixture, the function `DPdensity` from the `R` package `DPpackage` provides an off-the-rack solution. In practice, the MCMC scheme generates, at each iteration $s = 1, \ldots, S$, values $(\underline{w}^s, \underline{\mu}^s, \underline{\Sigma}^s)$ which, substituted in expression (4.20), produce a density function, $f^{BNJ,s}_{X^i,Y}$. Analytical expressions for the marginal and conditional densities, $f^{BNJ,s}_{Y}$ and $f^{BNJ,s}_{Y|X^i}$ as mixtures of normal distributions are made easily available by the choice of the Gaussian kernel. Clearly, it is also possible to evaluate the corresponding cumulative distribution functions. Thus, it is possible compute the global sensi-

tivity measures of interest, $\eta_i^{BNJ,s}$, $\delta_i^{BNJ,s}$, $\beta_i^{BNJ,s}$ from their definitions (Table 4.1), obtaining a posterior sample of each. We denote the sample means by $\widehat{\eta}_i^{BNJ}$, $\widehat{\delta}_i^{BNJ}$ and $\widehat{\beta}_i^{BNJ}$, respectively, proposing them as Bayesian point estimators. Approximate credibility intervals can be obtained from the empirical quantiles of the samples. The procedure is summarized in Section 4.5.2, to which we refer for further details.

It is important to observe that the known marginal distribution for X does not, in general, coincide with the marginal distribution for X derived from each $f_{X^i,Y}^{BNJ,s}$. Thus, by using only the joint density $f_{X^i,Y}$ to estimate the sensitivity measures, important information, standard in global sensitivity analysis is wasted. In fact, inference for conditional densities based in the joint model is known to be approximate (see e.g. Müller and Quintana, 2004). In the next section, we present an alternative estimation method which avoids this problem through a recent Bayesian approach to conditional density estimation.

4.4.2 Conditional density-based estimation

We now propose to use a Bayesian non-parametric regression model to do inference directly on the conditional density of $Y|X^i$, thus using all of the information contained in the *Data* to estimate the relationship between the variables and exploiting the knowledge of the marginal distribution of X to obtain the marginal distribution of Y. The idea is to transform the non-parametric mixture of equation (4.20) into a mixture of conditional densities:

$$f_{Y|X}(y|x) = \int \mathcal{K}(y|x,\theta)\mathrm{d}P_x(\theta), \tag{4.21}$$

This time a non-parametric prior, Π, is placed on the family, $\{P_x\}_{x\in\mathcal{X}}$ of mixing distributions indexed by x. Analogous to the DP mixture model of the previous section, a dependent DP mixture model or DDP mixture (MacEachern, 1999, 2000) is obtained when P_x follows a DP prior, marginally for every x, so that:

$$\mathbb{P}_x(\theta) = \sum_{\ell=1}^{\infty} w_\ell(x)\delta_{\theta_\ell(x)}. \tag{4.22}$$

The random covariate-dependent weights $W_\ell(x)$ follow the stick-breaking construction of Eq. (4.19), for i.i.d. random processes $\{V_\ell(x) : x \in \mathcal{X}\}$. In other words, $\mathbf{V}(x) \sim \mathcal{DP}$ for every x. It has been proved sufficient flexibility is achieved through models in which only the particles θ_ℓ or the weights w_ℓ depend on the covariate x (Barrientos et al., 2012), the second option being favoured due to better predictive capabilities. Several proposals have been studied in the literature, focusing on alternative definitions of the random functional weights $w_\ell(x)$ (e.g. Dunson and Park, 2008; Griffin and Steel, 2006; Rodriguez and Dunson, 2011).

The stick-breaking structure of the weights, which imposes a geometric decay, may be by-

passed through an alternative construction allowing further flexibility:

$$w_\ell(x) = \frac{\omega_\ell \mathcal{K}(x|\psi_\ell)}{\sum_{\ell'=1}^{\infty} \omega_{\ell'} \mathcal{K}(x|\psi_{\ell'})}. \tag{4.23}$$

The denominator of this expression is, again, an infinite mixture of parametric kernels, $\mathcal{K}$, this time with support $\mathcal{X}$. Each ω_ℓ can be interpreted as the probability that a realization of Y comes from the ℓ-th regression component regardless of the value of X, just as ω_ℓ is the conditional probability given $X = x$. Such density regression model, where the weights w_ℓ in (4.23) follow the stick-breaking representation of (4.19) and the extended parameters (θ_ℓ, ψ_ℓ) are i.i.d. from some adequate base measure, G, was proposed by Antoniano-Villalobos et al. (2014), to which we refer the reader for additional details on the role and choice of hyper parameters, as well as the algorithm used for inference.

We adopt this construction to estimate the conditional density $f_{Y|X^i}(y|x^i)$ as a mixture of linear regression models:

$$f_{Y|X^i}(y|x^i) = \sum_{\ell=1}^{\infty} w_\ell(x^i) \mathcal{N}(y|a_\ell + b_\ell x^i, \sigma_\ell), \tag{4.24}$$

where $w_\ell(x^i)$ is given by Eq. (4.23), with a DP prior. Once again, a MCMC approach is used to generate a sample of values, $(\underline{\theta}^s, \underline{\psi}^s) = (\underline{a}^s, \underline{b}^s, \underline{\sigma}^s, \underline{\omega}^s, \underline{\mu}^s, \tau^s)$, $s = 1 \dots S$, this time from the posterior distribution of $f_{Y|X^i}$. Each $f_{Y|X^i}^{BNC,s}(y|x^i), s = 1, \dots, S$, together with the known marginal for X^i can be used to calculate (e.g. by numerical integration) a corresponding marginal for Y. As discussed in Section 4.4.1, this is all that is needed to compute the global sensitivity measures of interest, $\eta_i^{BNC,s}$, $\delta_i^{BNC,s}$ and $\beta_i^{BNC,s}$. These, again allow point estimation, e.g. via the Monte Carlo averages, which we denote by $\widehat{\eta}_i^{BNC}$, $\widehat{\delta}_i^{BNC}$ and $\widehat{\beta}_i^{BNC}$, and interval estimation, via empirical quantiles. Section 4.5.3 summarizes the estimation procedure and offers additional technical details.

4.4.3 Simulation study

We examine the performance of the classes of partition-independent estimators proposed in Sections 4.4.1 and 4.4.2, via the 2−input and 21−input simulators introduced in Section 4.3.1.

For both joint and conditional density-based estimation, we set a burn-in period as $10n$ and the stored MCMC samples size $S = 1000$. Results are illustrated in Figures 4.5 and 4.6. Let us consider the 2−input simulator. In terms of ordinal ranking, Figure 4.5 suggests that both estimators correctly recover the equal importance of the two simulator inputs. The credibility intervals obtained from the joint model are wider than those for the conditional model, as expected, because additional uncertainty is introduced by neglecting to profit from the knowledge of the true marginal distribution of X^i. In terms of cardinal values, while deeming the inputs equally important, the joint estimators $\widehat{\eta}_i^{BNJ}$ and $\widehat{\beta}_i^{BNJ}$ overestimate the true value, while $\widehat{\delta}_i^{BNJ}$ is closer to it. The conditional estimators $\widehat{\eta}_i^{BNC}$, $\widehat{\delta}_i^{BNC}$ and $\widehat{\beta}_i^{BNC}$ correctly suggest

the simulator inputs as equally important, but underestimate the sensitivity measures, with the exact values falling outside of the credibility intervals, suggesting that stating definitive conclusions on the exact numerical values of the sensitivity estimates at such small sample sizes is risky.

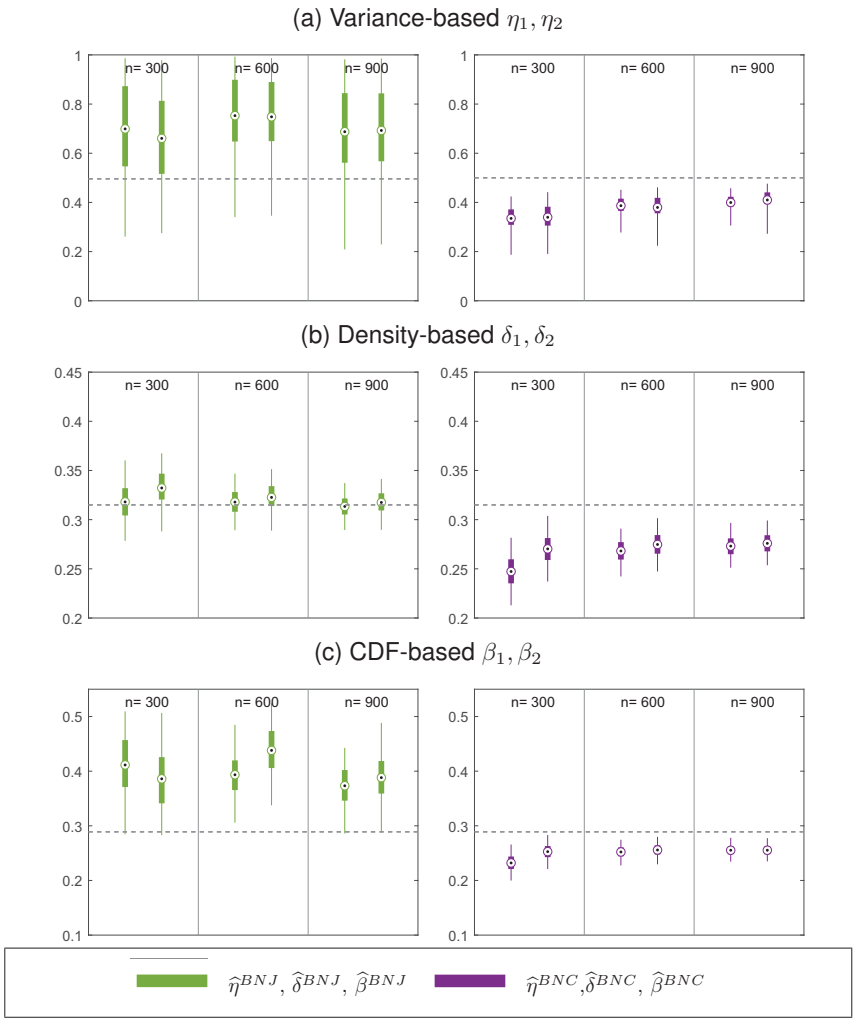


Figure 4.5: Results for input X^1, X^2 the 2−input simulator in Eq. (4.15): comparison of sensitivity measures estimates with 95% credibility intervals using Bayesian non-parametric partition-free joint/conditional estimators. The dash lines are analytical values of sensitivity measures in Table 4.2.

Figure 4.6 reports results for the 21−input simulator. The Bayesian non-parametric joint estimators $\widehat{\eta}_i^{BNJ}$, $\widehat{\delta}_i^{BNJ}$ and $\widehat{\beta}_i^{BNJ}$ correctly recover the true values of the parameters and, as the sample size increases from $n = 300$ to $n = 600$, the credibility intervals become narrower. At $n = 900$, there is no more overlap among the three groups of sensitivity measures, allowing the analyst to rank the inputs neatly. Regarding the Bayesian non-parametric conditional estimates, we also observe a reduction of the interval widths as the sample size increases. The important measures are overestimated, but ordinal ranking is accurately recovered, thus allowing the analyst to visualize that the first group of simulator inputs is more important than the second which,

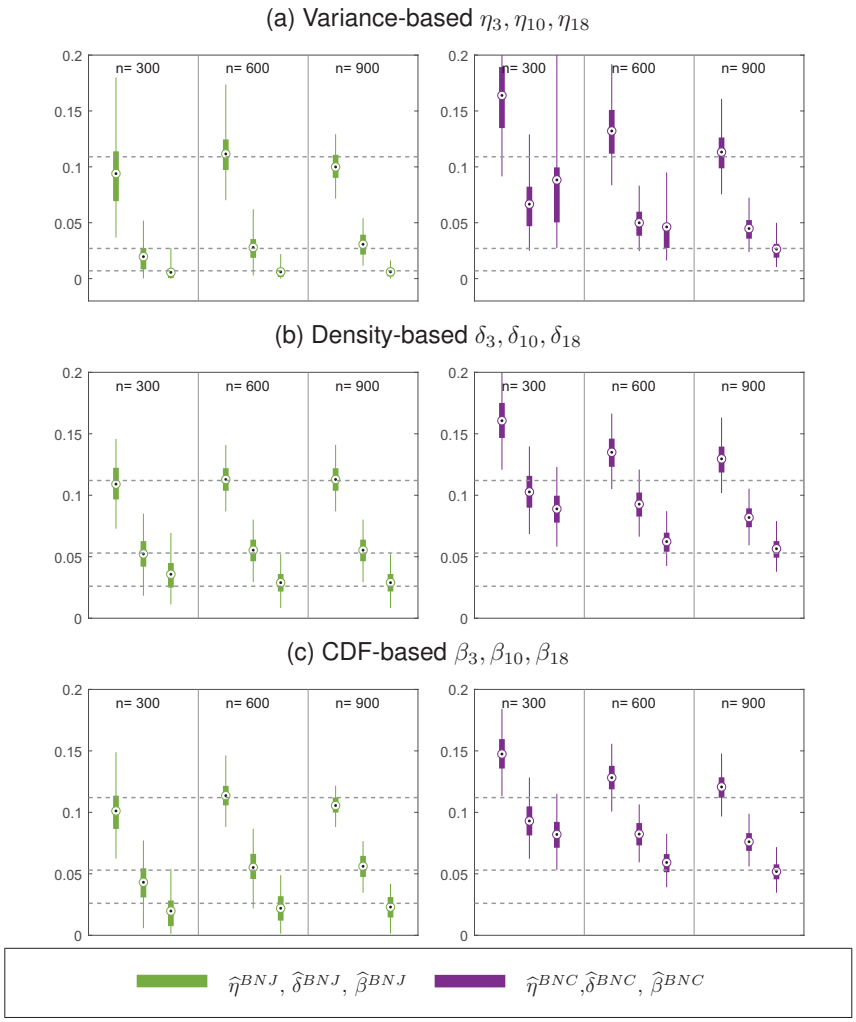


Figure 4.6: Results for input X^3, X^{10}, X^{17} of the 21−input simulator in Eq. (4.16): comparison of sensitivity measures estimates with 95% credibility intervals using Bayesian non-parametric partition-free joint/conditional estimators. The dash lines are analytical values of sensitivity measures in Table 4.2.

in turn, is more important than the third. For this example joint Bayesian estimators seem to outperform their conditional counterpart. This, again, is to be expected, since the joint Gaussian structure of the data is more easily recovered by the joint model in this case, so the loss due to ignoring the true distribution of X^i has a lesser effect on the results. However, we can appreciate a reassuring improvement in the estimation with larger sample sizes. One may argue that, in a situation in which the true conditional distribution of Y given X^i is unknown and may be complex, estimation based on the conditional density model may be preferred, as more robust; the price to pay is that a larger sample size may be required, specially in high-dimensional situations.

In Section 4.6, we test the behavior of the proposed methods when applied to a benchmark realistic application for sensitivity experiments.

4.5 Implementation details for the Bayesian non-parametric estimators

We present further details regarding the implementation of the Bayesian non-parametric estimation methods in Sections 4.3 and 4.4. Inference on the three selected sensitivity measures η_i, β_i and δ_i is performed independently for each $i = 1, \ldots, k$. Therefore, in order to simplify the notation, we will leave out the index i throughout this section, considering its value fixed. Throughout this section, all the integrals are approximated numerically using trapezoidal rule, and all the supremes are approximated by the maximum on a predetermined grid over $\mathcal{Y}$.

4.5.1 Partition-dependent bootstrap and Pólya urn estimation

Recall that in Section 4.3, given M, we have the partition $\{\mathcal{X}_m\}_{m=1}^M$ of $\mathcal{X}$ according to the sample proportion and corresponding $\{\mathbf{y}_m\}$.

Within each partition set, we generate $n - n_m$ new points $\widetilde{\mathbf{y}}_m^s$ and obtain the extended vector $\mathbf{y}_m^{C,s} = (\mathbf{y}_m, \widetilde{\mathbf{y}}_m^s)$ with $C \in \{Bb, Pu\}$, where $\widetilde{\mathbf{y}}_m^s$ is sampled from the posterior mean $\widetilde{G}_m$ for $C = Bb$, and is generated through Pólya urn scheme when $C = Pu$. The superscript s is used to indicate the $s-$th replicate.

After repeating the sampling procedure for S times, we obtain the partition-depended Bayesian estimator of η by calculating the Monte Carlo average:

$$\widehat{\eta}^C = \frac{1}{S}\sum_{s=1}^{S}\eta^{C,s}, \quad \text{with} \quad \eta^{C,s} = \sum_{m=1}^{M}\frac{n_m}{N}\frac{\left(\bar{y}_m^{C,s} - \bar{y}\right)^2}{s_y^2}, \tag{4.25}$$

where $\bar{y}_m^{C,s}$ is the sample mean of $\mathbf{y}_m^{C,s}$; $\bar{y}$ and s_y^2 are the sample mean and variance of $\mathbf{y}$. Approximate credibility intervals of η can be obtained from the empirical quantiles of $\{\eta^{C,s}, s = 1 \ldots, S\}$. The same intuition is used for δ and β. Specifically, we use

$$\widehat{\delta}^C = \frac{1}{S}\sum_{s=1}^{S}\delta^{C,s}, \quad \text{with} \quad \delta^{C,s} = \sum_{m=1}^{M}\frac{n_m}{N}\int_{\mathcal{Y}}|\hat{f}_Y^{\star}(y) - \hat{f}_m^{C,s}(y)|\mathrm{d}y, \tag{4.26}$$

$$\widehat{\beta}^C = \frac{1}{S}\sum_{s=1}^{S}\beta^{C,s}, \quad \text{with} \quad \beta^{C,s} = \sum_{m=1}^{M}\frac{n_m}{N}\sup_{y\in\mathbf{y}_m^{C,s}}\left|\hat{F}_Y(y) - \hat{F}_m^{C,s}(y)\right|, \tag{4.27}$$

where $\hat{f}_Y^{\star}$ and $\hat{f}_m^{C,s}$ are kernel smoothing functions of $\mathbf{y}$ and $\mathbf{y}_m^{C,s}$, respectively; $\hat{F}_Y$, and $\hat{F}_m^{C,s}$ are the empirical cdf's of $\mathbf{y}$ and $\mathbf{y}_m^{C,s}$, respectively.

Note that the calculations of $\eta^{C,s}$, $\delta^{C,s}$ and $\beta^{C,s}$ are equivalent to the pdf-based estimators in Eqs. (4.3), (4.4), (4.6) but with the enriched samples. Alternatively, the cdf-based estimators in Eqs. (4.8) and (4.5) could be used for $\eta^{C,s}$ and $\delta^{C,s}$.

4.5.2 Partition-free joint density-based estimation

Following the proposal in Jara et al. (2011), we fix $\alpha = 1$, and choose G to be a Normal-Inverse Wishart distribution

$$(\mu_\ell, \Sigma_\ell)|(m_1, \gamma, \psi_1) \overset{iid}{\sim} \mathcal{N}(\mu_\ell|m_1, \frac{1}{\gamma}\Sigma) IW(\Sigma_\ell|4, \psi_1), \quad \ell = 1, 2, \ldots,$$

where $\mathcal{N}(\cdot|m, A)$ denotes a bivariate normal distribution with mean m and covariance matrix A, and $IW(\cdot|4, \psi)$ denotes an Inverse-Wishart distribution with mean ψ^{-1}. A hyper-prior is assigned to the parameters of the base measure, with hyperparameters determined empirically:

$$\gamma \sim \text{Gamma}\left(\cdot|0.5, 0.5\right), \; m_1|(m_2, s_2) \sim \mathcal{N}(\cdot|m_2, s_2), \; \psi_1|(s_2) \sim IW(\cdot|4, s_2^{-1}),$$

where $\text{Gamma}(\cdot|a_1, a_2)$ denotes the Gamma distribution with mean a_1/a_2; $m_2 = (\mu_X, \bar{y})$ and $s_2 = \text{diag}(\sigma_X^2, s_y^2)$.

Inference is achieved through the function `DPdensity` from the `DPpackage` in `R`. The output is a MCMC posterior sample $\underline{\theta}^s = (\underline{w}^s, \underline{\mu}^s, \underline{\Sigma}^s)$, $s = 1, \ldots, S$. In practice, the number J_s of components with non-zero weights is finite, thus we have

$$\begin{aligned} \underline{w}^s = (w_1^s, \ldots, w_{J_s}^s), \quad \underline{\mu}^s &= (\mu_1^s, \ldots, \mu_{J_s}^s), \quad \underline{\Sigma}^s = (\Sigma_1^s, \ldots, \Sigma_{J_s}^s), \\ \text{with} \qquad \mu_\ell^s &= \begin{bmatrix} \mu_{1,\ell}^s \\ \mu_{2,\ell}^s \end{bmatrix}, \quad \Sigma_\ell^s = \begin{bmatrix} \sigma_{1,\ell}^s & \sigma_{3,\ell}^s \\ \sigma_{3,\ell}^s & \sigma_{2,\ell}^s \end{bmatrix}. \end{aligned} \tag{4.28}$$

Given the posterior realizations, the corresponding joint density can be obtained:

$$f_{X,Y}^{BNJ,s}(x, y|\underline{\theta}^s) = \sum_{\ell=1}^{J_s} w_\ell^s \cdot \mathcal{N}(x, y|\mu_\ell^s, \Sigma_\ell^s).$$

By the properties of the bivariate Normal distribution, the marginal and conditional distributions, $f_Y^{BNJ,s}$ and $f_{Y|X^i}^{BNJ,s}$ respectively, are also mixtures of Normal distributions:

$$f_Y^{BNJ,s}(y|\underline{\theta}^s) = \sum_{\ell=1}^{J_s} w_\ell^s \cdot \mathcal{N}(y|\mu_{2,\ell}^s, \sigma_{2,\ell}^s), \quad f_{Y|x}^{BNJ,s}(y|x, \underline{\theta}^s) = \sum_{\ell=1}^{J_s} w_\ell^s \cdot \mathcal{N}\left(\cdot|\nu_{2,\ell}^s, \tau_{2,\ell}^s\right) \tag{4.29}$$

where $\nu_\ell^s = \mu_{2,\ell}^s + \sigma_{3,\ell}^s(x - \mu_{1,\ell}^s)/\sigma_{1,\ell}^s$ and $\tau_\ell^s = \sigma_{2,\ell}^s - (\sigma_{3,\ell}^s)^2/\sigma_{1,\ell}^s$. Clearly, the corresponding cdfs, $F_Y^{BNJ,s}$ and $F_{Y|X}^{BNJ,s}$, as well as the marginal mean and variance can be calculated trivially. In particular,

$$\mu_Y^s \coloneqq \mathbb{E}[Y|\underline{\theta}^s] = \sum_{\ell=1}^{J_s} w_\ell^s \mu_{2,\ell}^s, \quad V_Y^s \coloneqq \mathbb{V}[Y|\underline{\theta}^s] = \sum_{\ell=1}^{J_s} w_\ell^s \left(\sigma_{2,\ell}^s + \left(\mu_Y^s - \mu_{2,\ell}^s\right)^2\right). \tag{4.30}$$

Thus, MCMC samples of the sensitivity measures of interest can be obtained as follows:

$$\eta^{BNJ,s} \approx \frac{V^s}{V_Y^s}; \quad \delta^{BNJ,s} \approx \frac{1}{2}\int_{\mathcal{X}}\int_{\mathcal{Y}} \left| f_{X,Y}^{BNJ,s} - f_X \cdot f_Y^{BNJ,s} \right| \mathrm{d}y\mathrm{d}x; \quad \beta^{BNJ,s} \approx \int_{\mathcal{X}} \sup_{\mathcal{Y}} \left| F_Y^{BNJ,s} - F_{Y|X}^{BNJ,s} \right| f_X \mathrm{d}x,$$

where

$$\begin{aligned} \mu_Y^s(x) &:= \mathbb{E}[Y|X=x,\underline{\theta}^s] = \sum_{\ell=1}^{J_s} w_\ell^s \nu_{2,\ell}^s, \\ V^s &= \int_{\mathcal{X}} \left(\mu_Y^s(x) - \mu_Y^s\right)^2 f_X dx = \int_{\mathcal{X}} \left(\sum_{\ell=1}^{J_s} w_\ell^s \frac{\sigma_{3,\ell}^s}{\sigma_{1,\ell}^s} \left(x - \mu_{1,\ell}^s\right) \right)^2 f_X \mathrm{d}x. \end{aligned}$$

Point estimators of interest are obtained as Monte Carlo averages:

$$\widehat{\eta}^{BNJ} = \frac{1}{S}\sum_{s=1}^{S} \eta^{BNJ,s}, \qquad \widehat{\delta}^{BNJ} = \frac{1}{S}\sum_{s=1}^{S} \delta^{BNJ,s}, \qquad \widehat{\beta}^{BNJ} = \frac{1}{S}\sum_{s=1}^{S} \beta^{BNJ,s}. \tag{4.31}$$

4.5.3 Partition-free conditional density-based estimation

Following the proposal of Antoniano-Villalobos et al. (2014), we fix $\alpha = 1$ and choose $\mathcal{K}(x|\psi_\ell)$ to be a Normal kernel, with $\psi_\ell = (\mu_\ell, \tau)$. The base measure G is given by:

$$\tau \sim \text{Gamma}(\cdot \mid 1,1); \quad (\mathbf{b}_\ell, \sigma_\ell, \mu_\ell) \overset{iid}{\sim} \mathcal{N}\left(\mathbf{b}_\ell \mid \mathbf{b}_0, \sigma_\ell C^{-1}\right) \text{Gamma}(\sigma_\ell^{-1} \mid 1,1) \mathcal{N}\left(\mu_\ell \mid \mu_0, (\tau/10)^{-1}\right),$$

where $\mathbf{b}_\ell = (a_\ell, b_\ell)$.The hyperparameters are chosen empirically. We consider the scatter-plot of $(\mathbf{x}, \mathbf{y})$ and the convex hull, i.e. the smallest convex set containing all points; we denote by the largest and smallest slopes a_{max}, a_{min} of the lines that constitute the convex hull, and the largest and smallest intercepts b_{max}, b_{min}. We then fix $\mathbf{b}_0 = (\bar{a}, \bar{b})$ where $\bar{a} = (a_{max} + a_{min})/2$ and $\bar{b} = (b_{max} + b_{min})/2$. We fix $C^{-1} = \text{diag}(\sigma_a^2, \sigma_b^2)$, where $\sigma_a = 1/3(a_{max} - \bar{a})$ and $\sigma_b = 1/3(b_{max} - \bar{b})$.

We use the MATLAB subroutine provided by Antoniano-Villalobos et al. (2014) to generate an MCMC posterior sample $(\underline{\theta}^s, \underline{\psi}^s) = (\underline{a}^s, \underline{b}^s, \underline{\sigma}^s, \underline{\omega}^s, \underline{\mu}^s, \tau^s)$, $s = 1 \ldots S$, where

$$\begin{aligned} \underline{a}^s &= (a_1^s, \ldots, a_{J_s}^s), \quad \underline{b}^s = (b_1^s, \ldots, b_{J_s}^s), \quad \underline{\sigma}^s = (\sigma_1^s, \ldots, \sigma_{J_s}^s), \\ \underline{\omega}^s &= (\omega_1^s, \ldots, \omega_{J_s}^s), \quad \underline{\mu}^s = (\mu_1^s, \ldots, \mu_{J_s}^s). \end{aligned} \tag{4.32}$$

Given the a posterior realization $(\underline{\theta}^s, \underline{\psi}^s))$, a conditional density can be obtained from

Eqs. (4.23) and (4.24):

$$f_{Y|X}^{BNC,s}(y|x,\underline{\theta}^s,\underline{\psi}^s)=\sum_{\ell=1}^{J_s}w_\ell^s(x)\mathcal{N}\left(y|a_\ell^s+b_\ell^s x,\sigma_\ell^s\right). \tag{4.33}$$

The corresponding marginal pdf $f_Y^{BNC,s}$ of Y is obtained by integrating with respect to the true f_X:

$$f_Y^{BNC,s}(y|\underline{\theta}^s)\approx\int_{\mathcal{X}}f_{Y|X}^{BNC,s}f_X\mathrm{d}x. \tag{4.34}$$

Clearly, the corresponding marginal and conditional cdfs, $F_{Y|X}^{BNC,s}$ and $F_Y^{BNC,s}$, respectively can be obtained trivially. In particular, posterior realizations of the marginal mean and variance of Y are given by

$$\mu_Y^s\coloneqq\mathbb{E}[Y|\underline{\theta}^s,\underline{\psi}^s]\approx\int_{\mathcal{Y}}yf_Y^{BNC,s}\mathrm{d}y,\quad V_Y^s\coloneqq\mathbb{V}[Y|\underline{\theta}^s,\underline{\psi}^s]\approx\int_{\mathcal{Y}}\left(y-\mu_Y^s\right)^2f_Y^{BNC,s}\mathrm{d}y \tag{4.35}$$

Thus, MCMC samples of the sensitivity measures of interest can be obtained as follows:

$$\eta^{BNC,s}\approx\frac{V^s}{V_Y^s};\quad\delta^{BNC,s}\approx\frac{1}{2}\int_{\mathcal{X}}\int_{\mathcal{Y}}\left|f_Y^{BNC,s}-f_{Y|X}^{BNC,s}\right|\mathrm{d}yf_X\mathrm{d}x;\quad\beta^{BNC,s}\approx\int_{\mathcal{X}}\sup_{\mathcal{Y}}\left|F_Y^{BNC,s}-F_{Y|X}^{BNC,s}\right|f_X\mathrm{d}x,$$

where

$$\mu_Y^s(x)\coloneqq\mathbb{E}[Y|x,\underline{\theta}^s,\underline{\psi}^s]=\sum_{\ell=1}^{J_s}\omega_\ell^s(x)\left(a_\ell+b_\ell x\right).$$

$$\widetilde{\mu}_Y^s\coloneqq\mathbb{E}[\mu_Y^s(X)]\approx\int_{\mathcal{X}}\mu_Y^s(x)f_X\mathrm{d}x,\quad V^s=\mathbb{V}[\mu_Y^s(X)]\approx\int_{\mathcal{X}}\left(\mu_Y^s(x)-\widetilde{\mu}_Y^s\right)^2f_X\mathrm{d}x.$$

Finally, point estimators of interest are obtained as Monte Carlo averages:

$$\widehat{\eta}^{BNC}=\frac{1}{S}\sum_{s=1}^{S}\eta^{BNC,s},\qquad\widehat{\delta}^{BNC}=\frac{1}{S}\sum_{s=1}^{S}\delta^{BNC,s},\qquad\widehat{\beta}^{BNC}=\frac{1}{S}\sum_{s=1}^{S}\beta^{BNC,s}. \tag{4.36}$$

Table 4.4: Simulator inputs for the LevelE code. $U(\cdot,\cdot)$ and $LU(\cdot,\cdot)$ stand for the uniform and log-uniform distributions respectively

Input	Meaning	Distribution
X_1	Containment time	U(100,1000)
X_2	Iodine Leach rate	LU$(10^{-3}, 10^{-2})$
X_3	Neptunium chain Leach rate	LU$(10^{-6}, 10^{-5})$
X_4	Iodine retention factor (1st layer)	LU$(10^{-3}, 10^{-1})$
X_5	Geosphere water velocity 1st layer	U(100,500)
X_6	Geosphere Length 1st layer	U(1,5)
X_7	Factor to compute Neptunium retention coefficients Layer 1	U(3,30)
X_8	water velocity in geosphere's 2nd layer	LU$(10^{-2}, 10^{-1})$
X_9	Length of geosphere's 2nd layer	U(50,200)
X_{10}	Retention factor for I (2nd layer)	U(1,5)
X_{11}	Factor to compute Neptunium retention coefficients Layer 2	U(3,30)
X_{12}	Stream flow rate	LU$(10^5, 10^7)$

4.6 Case study: LevelE simulator

In this section, we evaluate the performance of the proposed estimators through the benchmark simulator of sensitivity analysis, LevelE. The LevelE code simulates the release of radiological dose from a nuclear waste disposal site to humans over geological eras. The code has been developed in an international exercise launched by the Nuclear Energy Agency (NEA) in the mid 1980's (Nuclear Energy Agency, 1989). Goal of the exercise was the realization of a reference simulator for the prediction of flow and transport of radionuclides in actual geologic formations against which to compare other simulators developed internationally to support the selection of radioactive waste management policies. Since then, LevelE has become the benchmark simulator of sensitivity analysis (Saltelli et al., 2000; Saltelli and Tarantola, 2002). During the international exercise, distributions for the uncertain simulator inputs were assessed (Table 4.4), and have become the reference for analysis on this code. From a technical viewpoint, the LevelE code solves of a set of nested partial differential equations that compute the released radiological dose in Sievert/year over a time range of $t = 10,000$ to 2×10^9 years. The detailed equations of the code are reported in Saltelli and Tarantola (2002).

Previous works have discussed the sensitivity analysis of this simulator using alternative sampling methods and sizes. For instance, Saltelli et al. (2000) employ $3,084$ simulator evaluations to obtain point estimates of the first and total order variance-based sensitivity indices. Saltelli and Tarantola (2002) employ $10,000$ simulator runs for the point estimation of first-order variance-based sensitivity indices, a second experiment with $16,384$ runs for the point of the first and total order sensitivity indices according to the design in Saltelli (2002a) (no uncertainty in the estimates is provided). In Ratto et al. (2007), stable patterns for the estimation of variance-based sensitivity measures are ob-

tained at a cost of about $1,024$, after the input-output dataset has been used to train an emulator. In Castaings et al. (2012), design based on substituted columns sampling and permuted columns sampling are used, with convergence at about 10^4 runs. Wei et al. (2014) propose a copula-based estimation methods that reduces the cost to about $1,000$ runs for point estimates, with 20 replicates for obtaining confidence intervals. Plischke and Borgonovo (2017) apply a given-data design for the point estimators $\widehat{\eta}_i^{\diamond}$, $\widehat{\delta}_i^{\diamond}$ and $\widehat{\beta}_i^{\diamond}$ using a sample up to size $n = 5,000$, with estimates becoming stable for $n > 1,000$ runs. Thus, a sample of size $n = 1,000$ can be considered reflective of state of art for the identification of the key-uncertainty drivers of LevelE.

We report results for the calculation of global sensitivity measures using all classes of estimators discussed in the present work for samples of sizes $n = 600$ and $n = 900$. Figures 4.7 and 4.8 display the results.

The graphs in Figure 4.7 report the Bayesian bootstrap and Pólya urn estimators, vis-á-vis the point estimators for variance-based (graphs in row *a*), density-based (graphs in row *b*) and cdf-based (graphs in row *c*) sensitivity measures. The results show that already at $n = 600$ the two most important simulator inputs are correctly identified. However, the estimates are sensitive to the partition size. Consider the right graph in row *a)*. The credibility intervals of the variance-based Pólya urn estimators with $M = 26$ are completely overlapping. This signals that, had the analyst chosen such partition size, the estimates would not be meaningful. The separation becomes, instead, clearer at smaller partition sizes with $M = 9$ being possibly the optimal choice. Note that the estimates tend to be upward biased as the partition size increases, in agreement with our previous experiments and also with previous literature findings.

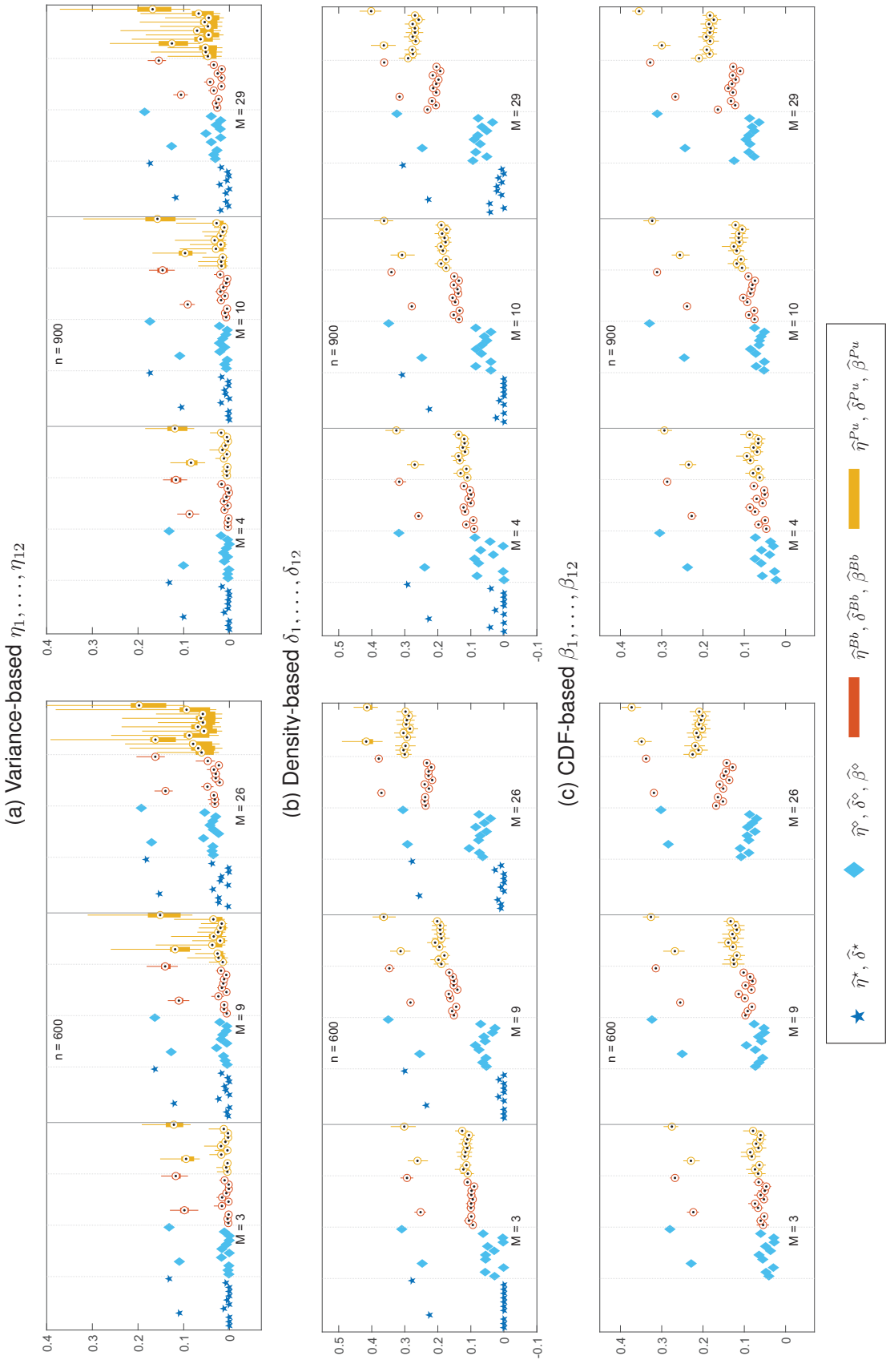


Figure 4.7: Results for the LevelE code: comparison of sensitivity measures estimates using frequintist pdf/cdf-based estimators and Partition-dependent Bayesian non-parametric estimators. Bayesian estimates include 95% credibility intervals.

We then come to the joint and conditional partition-independent Bayesian density estimators (Figure 4.8).

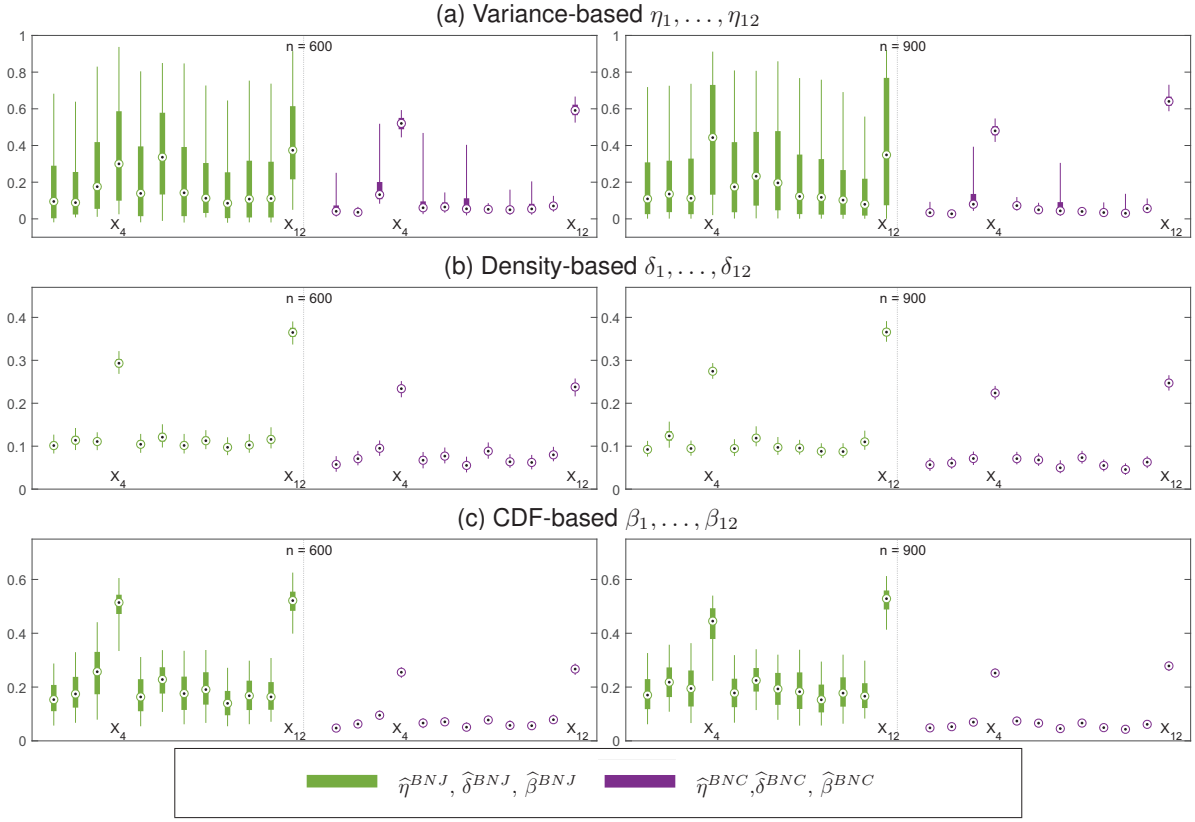


Figure 4.8: Results for the LevelE code: comparison of sensitivity measures estimates with 95% credibility intervals using Bayesian non-parametric partition-free joint/conditional estimators.

The two graphs in row *(a)* display the estimates and credibility intervals for variance-based sensitivity measures ($\widehat{\eta}_i^{BNJ}$,$\widehat{\eta}_i^{BNC}$), the two graphs in row *(b)* for density-based sensitivity measures ($\widehat{\delta}_i^{BNJ}$, $\widehat{\delta}_i^{BNC}$) and the two graphs in row *(c)* for cdf-based ($\widehat{\beta}_i^{BNJ}$, $\widehat{\beta}_i^{BNC}$) sensitivity measures. Figure 4.8 shows that the two key-uncertainty drivers are correctly identified already at $n = 600$, by $\widehat{\eta}_i^{BNC}$, $\widehat{\delta}_i^{BNC}$ and $\widehat{\beta}_i^{BNC}$, as the credibility intervals of the associated sensitivity measures separate from the credibility intervals of the remaining simulator inputs. The estimators $\widehat{\eta}_i^{BNJ}$ (based on joint Bayesian density estimation) do not produce meaningful results for variance-based sensitivity measures at either sample sizes. However, the estimators $\widehat{\delta}_i^{BNJ}$ and $\widehat{\beta}_i^{BNJ}$ correctly identify the two most influential simulator inputs.

Let us consider the perspective of an analyst interpreting the results overall. From the available *Data*, the analyst is able to obtain alternative estimators for representatives of three categories of sensitivity measures, with display of credibility intervals. With the exception of $\widehat{\eta}_i^{BNJ}$, the estimators communicate that uncertainty in the simulator response is mostly driven by two simulator inputs, with the remaining ones being of lower signifi-

cance. Thus, the analyst is allowed to confidently report the key-uncertainty drivers to the decision-maker even if the sample size is limited. At the same time, Figures 4.7 and 4.8 communicate that the sample is not sufficient to rank the medium and low-important simulator inputs with confidence. If the decision-maker (modeler) wished sharper estimates of the sensitivity measures of these inputs, the analyst would need a larger sample size. This could be obtained either through additional runs of the original simulator or by fitting an emulator and, in case the fit is accurate, running the emulator instead of the original code.

4.7 Discussion

This work has presented a fully Bayesian approach to the estimation of probabilistic sensitivity measures from a given sample. The proposed algorithms yield credibility intervals for the estimates without increasing computational burden. We have studied four classes of estimators. The first two find their theoretical ground in non-parametric Bayesian estimation based on the Dirichlet process. These estimators run in parallel with one-sample frequentist estimators currently in use, produce uncertainty in the estimates and are computationally simple to implement. However, they leave the analyst with the problem of choosing the optimal partition. The introduced conditional and unconditional non-parametric Bayesian estimators eliminate the partition selection problem, while producing uncertainty in the estimates. However, their numerical implementation needs to be carefully executed, as it requires a combination of numerical integration and MCMC. Algorithms are available, but their convergence might take a longer time than the Bayesian bootstrap and Pólya urn estimators. Then, how should one proceed in a practical situation? The several numerical experiments performed by the authors (of which a subset was reported in the paper) evidence that the estimators succeed in identifying key-uncertainty drivers at small sample sizes in most situations. Then, a suggested approach would be to apply first the Bayesian bootstrap and/or Pólya urn estimators on the available sample for computing an ensemble of sensitivity measures (e.g., η, δ, β). If the sensitivity measure estimates and credibility intervals yield a clear picture of the simulator inputs influence, then the analysis could be considered satisfactory. However, the analyst ought to test this assertion repeating the estimates at alternative partition sizes. In case results are strongly dependent on the partition size, the analyst can invest in the Bayesian non-parametric estimation. If these estimators yield a clear picture about the simulator input influence, the analysis is conclusive. Conversely, a larger sample is needed and the analyst ought to plan for additional simulator runs.

While we have discussed three well-known global sensitivity measures, the paradigm presented here can be applied to the estimation of any global sensitivity measure, includ-

ing, among others, value of information, sensitivity measures based on any discrepancy between densities or cumulative distribution functions.

From a more general perspective, the work shows that combining recent advances in Bayesian non-parametric density estimation with probabilistic sensitivity analysis in DACE may lead to improvements in the estimation of global sensitivity measures. Research in Bayesian non-parametric density estimation is active in Statistics and Machine Learning, but the advances in this discipline are not directly known to the DACE community. This work represents a first systematic bridge between these two closely related areas of Statistics, and we hope it could favour further research for transferring findings in Bayesian-non parametric estimation to the field of computer experiments. At the same time, exposing Bayesian estimation to the demands coming from probabilistic sensitivity analysis of realistic simulators may challenge state of the art and stimulate further research in Bayesian-non parametric estimation.

Chapter 5

Kriging for large scale simulators

Kriging is one of the most widely used emulation methods in simulation. However, memory and time requirements may prevent its application to datasets generated by dimensionally large simulators. In this chapter, we merge Kriging with a recent innovation of the machine learning literature to propose a new algorithm that, while preserving prediction accuracy, notably reduces Kriging time and memory requirements. We analyze theoretically the error and prove that the algorithm lowers computational complexity of one order of magnitude. We then test the implementation of the algorithm in a series of challenging numerical experiments. To evaluate its performance, we use not only traditional measure of the prediction performance (e.g., root mean square error), but challenge the prediction ability through the computation of complex functionals of the simulator output. The proposed algorithm is compared against four Kriging subroutines, which are either commonly used in the operational research field or designed for large datasets. Experiments on simulators of increasing dimensionality show that the proposed algorithms offers significant improvements in time and memory and allows one to breach the $10,000$ *simulator input barrier for the first time.*
This chapter contains joint work with Alessandro Rudi, Emanuele Borgonovo and Lorenzo Rosasco, and will be submitted for publication shortly.

5.1 Motivation

The continuous growth in computing capabilities and the data driven revolution are making computational modeling and simulation experiments increasingly relevant for enterprises and decision makers. They allow us *to extract value from data and ask questions about behaviors; and then use the answers to understand, design, manage and predict the workings of complex systems and processes...* (UK Government Office for Science, 2018, p. 6). However, while the steady increase in computing capabilities allows analysts to build simulators of increasing complexity and sophistication (Luo et al., 2015), *simulation*

models are often tedious to build, need substantial data for input modeling, and require significant time to run(Ankenman et al., 2010). Thus, computational burden might be an obstacle to fully exploit the capabilities of simulation codes. Relevant research efforts are devoted to increase computational efficiency (Luo et al., 2015). One main strategy for fully exploiting the insights of a simulator is to substitute the time-consuming computer code with a fast-running emulator on a region of interest.

Kriging (also called Gaussian process regression in statistical context) is one of the most well known and applied emulation methods in simulation (Ankenman et al., 2010; Chen et al., 2013). The merits of Kriging are several, ranging from its analytical tractability For example, [1] to the fact that an analyst obtains not only a point estimate, but a corresponding estimate of the prediction error (Santner et al., 2003). Three aspects are of primary interest to the analyst : 1) accuracy, 2) the ability of using all available information, 3) the speed in training and prediction. The first is essential for obvious reasons. The second often generates a trade-off, insofar exploiting available information is key for increasing prediction accuracy, but large datasets make Kriging time and memory consuming. Specifically, in training, memory requirements associated with matrix inversion limit the size of the input-output dataset that can be used. In prediction, the execution time increases with the matrix dimension reducing the convenience of using the emulator.

Our goal is to introduce an algorithmic approach that reduces Kriging memory and time requirements, thus allowing the application of Kriging to simulators of larger dimensionalities or to input-output samples with larger sizes than in current practice. We merge the regularization algorithms of the Machine learning literature and the typical algorithmic implementation of Kriging in simulation. In the proposed algorithm, the part concerning the predictive mean borrows from the work of Rudi et al. (2015) on Nyström regularization. The approach is then modified to take into account the constraint of positive variance, which is often neglected in Machine Learning studies. We provide a theoretical analysis, obtaining an expression for the predictive variance and showing that the algorithm requires a total cost of

$$O(nm(d+m)) \text{ in time}, \quad O(m^2+n) \text{ in space}.$$

Note that universal Kriging algorithms requirements are of $O(n^3)$ in terms of time and space. We call the proposed approach 'fast Kriging'.

We then challenge the proposed fast Kriging in a series of numerical experiments for simulators of increasing dimensionality. To test its performance, we use not only traditional metrics such as the root mean square error (RMSE), but also use functionals of the output. In particular, the intuition is that the accurate estimation of a complex functional

[1] Oakley and O'Hagan (2004) investigate the analytical tractability of Kriging in application of sensitivity analysis, especially under Gaussian input distribution assumption.

of the output is a sign of how accurately the emulator forecasts the output distribution. Moreover, the estimation of a complex functional requires thousands of predictions, serving as a test of whether the algorithm produces a time advantage in prediction. As functionals, we use global sensitivity measures based on alternative distance metrics that require an accurate estimation of the conditional and unconditional distributions of the model output. The details of global sensitivity measures are given in Sections 2.2.3, 2.2.4 and 2.2.5 of Chapter 2.

In the experiments, the algorithm is compared against four Kriging implementations available in the literature. In particular, we use the well-known MATLAB toolbox called 'DACE', developed by Lophaven et al. (2002), and the Kriging implementation in UQlab, a MATLAB metamodelling tool created by Lataniotis et al. (2015). The third and fourth subroutines belong to the R package `laGP`, developed by Gramacy (2016) to address large datasets (see Section 5.3 for further details).

The experiments are carried out with simulators of increasing dimensionality starting with a synthetic test case, then moving to the LevelE code, a benchmark simulator well known for its low numerical tractability. We then move to the STOCFOR3 linear program, the largest linear program in the online-available NETLIB library, with $40,000$ uncertain simulator inputs.

Overall, the experiments show that the proposed algorithm notably reduces memory and time requirements while maintaining the same level of accuracy. Also, for the STOCFOR3 problem, the proposed implementation is the only to grant the emulation of the original simulator, as the other four subroutines fail due to memory or time requirements.

This chapter is organized as follows. Section 5.2 clarifies the notation used in this chapter. Section 5.3 introduces the background and relevant literature of Kriging. Section 5.4 is devoted to derive the proposed fast Kriging technique and offers the corresponding theoretical analysis. Section 5.5 presents the numerical results. Section 5.8 provides a brief summary.

5.2 Notation

We first clarify the notation used in this chapter. One lets $y = g(\mathbf{x})$ denote the simulator input-output mapping, where y represents the output observation, and $\mathbf{x} = (x^1, \dots x^d)^\top \in \mathcal{X} \subseteq \mathbb{R}^d$ is the corresponding d-dimensional input vector [2].

The idea of meta-modeling is to use an explicit and simple function g_{meta} to replace the original simulator g, that is $y = g_{meta}(\mathbf{x})$, where g_{meta} gives an approximation of the

[2]Note that in the previous chapters, the number of inputs is denoted as k. In this chapter, k is used for kernel functions.

Table 5.1: Some symbols and notation used in this chapter. Matrices are capitalized and vectors are in bold.

Symbol	Meaning
d	dimension of input space $\mathcal{X}$
$\mathcal{GP}$	Gaussian process
$k(\mathbf{x}, \mathbf{x}')$	covariance (or kernel) function evaluated at $\mathbf{x}$ and $\mathbf{x}'$
K, $K(X, X)$	$n \times n$ covariance matrix
$r(\mathbf{x}, \mathbf{x}')$	correlation function evaluated at $\mathbf{x}$ and $\mathbf{x}'$
R, $R(X, X)$	$n \times n$ correlation matrix
$\mathbb{P}$	a measure
n and n_{train}	number of training cases
n^* and n_{pred}	number of test cases
X	$d \times n$ matrix of the training inputs $\{\mathbf{x}_j\}_{j=1}^n$: the design matrix
$\mathbf{x}_j$	the j-th training input

output response surface. Our goal is to make inference on g_{meta} based on a given dataset (the training set) and provide a prediction of the output when given a new input point (or testing/prediction point).

We denote the training set by $Data = \{(\mathbf{x}_j, y_j)\}_{j=1}^n$, which consists of n input-output pairs. We aggregate the n input vectors in a $d \times n$ design matrix $X = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$, and arrange the output observations in a column vector $\mathbf{y} = (y_1, \ldots, y_n)^\top$. Table 5.1 summarizes some notation used in this chapter.

5.3 Background

Kriging originates in geostatistics and spatial statistics as an exact interpolation method (Krige, 1952). Subsequent works such as Matheron (1975); Howarth (1979); Welch et al. (1992); Ver Hoef and Cressie (1993); Morris (1993) elaborate the mathematical properties of the method.

Following Sacks et al. (1989), the so-called universal Kriging emulator (also known as Gaussian process regression in Santner et al. (2003)) is constructed as:

$$y = \phi(\mathbf{x})^\top \mathbf{w} + \mathcal{M}(\mathbf{x}), \tag{5.1}$$

where

- $\phi(\mathbf{x}) = (\phi_1, \ldots, \phi_p)^\top$ is a fixed basis function which maps a $d-$dimensional input vector $\mathbf{x}$ into a $p-$dimensional feature space;
- $\mathbf{w}$ is a size p column vector of weights (regression coefficients), and the term $\phi(\mathbf{x})^\top \mathbf{w}$ is called *trend*;

- $\mathcal{M}(\mathbf{x})$ is a stationary Gaussian process with zero mean, $\mathbb{E}[\mathcal{M}] = 0$. In general, the covariance between outputs is assumed to be proportional to a correlation matrix,

$$cov\left(\mathcal{M}(\mathbf{x}), \mathcal{M}(\mathbf{x}')\right) = \sigma^2 r(\mathbf{x}, \mathbf{x}'), \quad \mathbf{x}, \mathbf{x}' \in \mathcal{X} \tag{5.2}$$

  where $r(\mathbf{x}, \mathbf{x}')$ is the correlation function of input vectors. One typically assumes that the correlation function is separable, writing

$$r(\mathbf{x}, \mathbf{x}'|\theta) = \prod_{i=1}^{d} r(h_i|\theta_i), \quad \text{where} \quad h_i = |x_i - x_i'|, \quad \theta = (\theta_1, \ldots, \theta_d). \tag{5.3}$$

  Williams and Rasmussen (2006, p. 80) present some commonly used correlation functions.

Kriging also shares a Bayesian interpretation (Koehler and Owen, 1996; Santner et al., 2003), where the response surface is regarded as a realization of a stochastic process and the mean of the posterior process is used as the predictor (Kleijnen, 2017).

An active research area of Kriging is the determination of efficient (or space-filling) designs. Chen et al. (2006); Kleijnen (2014) and Silvestrini et al. (2013) provide overviews of the use of orthogonal arrays and the maximum entropy principle for the selection of the design points. Sequential designs based on Kriging meta-models are presented in Kleijnen (2005); Huang et al. (2006); Kleijnen (2009, 2017).

Kriging for simulators with a stochastic response (stochastic kriging, henceforth) has been widely studied (Mitchell and Morris, 1992; Van Beers and Kleijnen, 2003; Forrester et al., 2008; Yin et al., 2009; Ankenman et al., 2010; Chen et al., 2012; Dellino et al., 2012; Picheny et al., 2013). In stochastic Kriging, researchers face the challenge of approximating the trend while simultaneously accounting for extrinsic and intrinsic noise (Ankenman et al., 2010). Extrinsic noise is the variability associated with the randomness of the Gaussian process. Intrinsic noise or 'nugget effect' in spatial statistics (Ver Hoef and Cressie, 1993, Chapter 3) is the variability associated with the randomness of the simulator response itself. Ankenman et al. (2010) develop an efficient stochastic Kriging meta-model that has become a benchmark for subsequent studies. Chen et al. (2013); Qu and Fu (2014) incorporate gradient information in stochastic Kriging. The intuition is to make use of information coming from calculating the gradients of Kriging to improve prediction accuracy or enforce known properties of the original model on the response surface (e.g., monotonicity). Chen et al. (2012) study the effect of common random numbers on stochastic Kriging showing that common random numbers worsen prediction performance, but improve the estimation of slope parameters and gradients. For other stochastic Kriging variants, we recall the works of Van Beers and Kleijnen (2003); Kleijnen and Van Beers (2005); Marrel et al. (2012); Yin et al. (2009, 2011), among others.

Xu (2012); Sun et al. (2014) develop an adaptive algorithm based on stochastic Kriging for optimisation problem with a large feasible set, where the intuition is to use efficient sampling scheme for sampling the next solution. Preuss et al. (2012) investigate the performance of Kriging-based optimization techniques for relatively high-dimensional problems ($d = 22$). We refer to Chen and Kim (2016); Barton and Meckesheimer (2006); Hong et al. (2015); Picheny et al. (2013); Jalali et al. (2017) for further details about Kriging-based optimization.

Works closely related to ours concern the research for reducing computational complexity. In fact, the estimation of Kriging parameters, i.e. θ and σ^2 in Eq. (5.1), is recognized as a mathematically challenging task, see Erickson et al. (2018) for a recent overview. In fact, the use of full-order kriging involves the inversion of a $n \times n$ matrix (see Section 5.4 for details). The memory requirement for storing the matrix is $O(n^2)$, while the computational time for the inversion is $O(n^3)$. For large n, the estimation becomes prohibitive.

Two main intuitions have been explored to reduce the computational complexity in the literature. The first is to make the Kriging prediction locally dependent on the training points in a neighborhood. For example, Urtasun and Darrell (2008) propose an on-line meta-model (assuming the prediction points arrive sequentially) based on a local mixture of Gaussian Processes. When a new point arrives, the algorithm first defines the training points which are located in the nearest neighborhood of this new point, then calculates the prediction based on those selected training points. Thus, the dimension n of the matrix to be inversed is limited. Gramacy and Apley (2013) use a similar idea and apply different criteria to determine the sub-design region such as active learning Cohn (ALC) and mean-square prediction error. This method has been implemented in the R package `laGP`, used in this work for comparison.

The second is to select a subset of the training dataset of size $m < n$, the so-called active set, the associated approaches are typically encountered in the machine learning literature. Entropy minimization and gradient-based optimization are frequently used in the selection of the active set. For instance, Lawrence et al. (2003) propose Gaussian process regression associated with a forward greedy search of the training points based on differential entropy score. Snelson and Ghahramani (2006) propose a Gaussian process regression with reduced covariance matrix parameterized by pseudo-input points, where the pseudo points are selected via gradient-based optimization. Hensman et al. (2013) fit a Gaussian process regression by using stochastic variational inference so that the resulting fitted surface only depends on a set of inducing variables. We refer to Quiñonero-candela et al. (2005) for a comprehensive review.

In this work, we borrow from ideas in Rudi et al. (2015), where Nyström regularization is used for randomly sampling the active set from the available dataset. The intuition of a regularization is that a suitable choice of the kriging scale parameter allows one to

capture the relevant information in the empirical kernel matrix. Nyström regularization achieves such goal by selecting a subset of columns, thus allowing one to manipulate and store only a fraction of the empirical kernel matrix. More specifically, in Nyström regularization the empirical kernel matrix is used together with a regularization term. The regularization term has the purpose of retaining the part of the matrix that contains the relevant information required by the algorithm, while discarding the part that is less relevant. See Appendix 5.4.1 and the next section for greater detail.

This concise literature review shows that Kriging is a widely used and studied methodology in the fields of simulation and machine learning. However, recent advances in the latter field have not crossed the disciplinary barrier. We believe that such crossing could benefit the use and application of Kriging in the simulation community. In the next section, we propose a methodology based on such cross-fertilization.

5.4 Methodology

5.4.1 Regularization

The first step for making Kriging faster is to relate Kriging and the so-called regularization methods of machine learning (Williams and Rasmussen, 2006). A central role is played by the concept of Reproducing Kernel Hilbert Space (RKHS). Let $f : \mathcal{X} \mapsto \mathbb{R}$, and $f \in \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space. $\mathcal{H}$ is called a RKHS with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$, if there exists a function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that: 1) $\forall \mathbf{x} \in \mathcal{X}, k(\mathbf{x}', \mathbf{x})$ is a function of $\mathbf{x}$ belonging to $\mathcal{H}$; and 2) k possesses the reproducing property, i.e., $\langle f(\cdot), k(\cdot, \mathbf{x}) \rangle = f(\mathbf{x})$. The function k is called reproducing kernel and the matrix $K = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^n, \forall \mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$ is positive semi-definite (Williams and Rasmussen, 2006). One can express the RKHS as [3]

$$\mathcal{H}_n = \left\{ f \in \mathcal{H} | f(\mathbf{x}) = \sum_{j=1}^{n} \alpha_j k(\mathbf{x}_j, \mathbf{x}), \mathbf{x}_j \in \mathcal{X}, \alpha_j \in \mathbb{R} \right\} \tag{5.4}$$

By the Moore-Aronszajn theorem (Aronszajn, 1950), the RKHS uniquely determines k and the converse, see Wendland (2004) for detailed discussions.

The problem then becomes to infer an underlying function $f(\cdot)$ from a finite dataset *Data*. One considers the functional

$$J[f] = Q(\mathbf{y}, \mathbf{f}) + \beta \|f\|_{\mathcal{H}}^2, \tag{5.5}$$

where $\|f\|_{\mathcal{H}}^2 = \langle f, f \rangle_{\mathcal{H}}$; in Eq. (5.5), the first term assesses the quality of data-fit (a

[3] The associated inner product is $\langle f, f' \rangle_{\mathcal{H}} = \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha'_j k(\mathbf{x}_i, \mathbf{x}'_j)$ where $f'(\mathbf{x}) = \sum_{j=1}^n \alpha'_j k(\mathbf{x}'_j, \mathbf{x})$.

measure between the observed y_i and fitted value $f(\mathbf{x}_i)$), and the second term is called *regularizer*.

The *representer theorem* of Kimeldorf and Wahba (1971) states that any minimizer $f \in \mathcal{H}$ of $J[f]$ has the form $f(\mathbf{x}) = \sum_{j=1}^{n} \alpha_j k(\mathbf{x}_j, \mathbf{x})$. Furthermore, if $J[f]$ is convex, the minimizer f is unique (Schölkopf et al., 2002).

A classical way to derive the minimiser solution is to consider a *Tikhonov regularization* approach:

$$\min_{f \in \mathcal{H}} \left(\frac{1}{n} \sum_{j=1}^{n} (y_j - f(\mathbf{x}_j))^2 + \beta \|f\|_{\mathcal{H}}^2 \right), \qquad \beta \geq 0. \tag{5.6}$$

The solution $\widehat{f}_\beta$ to Eq. (5.6) can be written as (Rudi et al., 2015):

$$\widehat{f}_\beta(\cdot) = \sum_{j=1}^{n} \hat{\alpha}_j k(\mathbf{x}_j, \cdot) \qquad \text{with} \qquad \widehat{\alpha} = (K + n\beta I_n)^{-1} \mathbf{y}, \tag{5.7}$$

where $\widehat{\alpha} = (\hat{\alpha}_1, \ldots, \hat{\alpha}_n)^\top$. One can re-write the above regularization predictor as

$$y^* = k(X, \mathbf{x}^*)^\top (K + n\beta I_n)^{-1} \mathbf{y}. \tag{5.8}$$

If one regards the kernel matrix K as the covariance matrix of output observations (recall that every covariance matrix is positive semi-definite), and imposes the covariance assumption in Eq. (5.2), one obtains:

$$y^* = R_*^\top \left(R + \frac{n\beta}{\sigma^2} I_n \right)^{-1} \mathbf{y}, \tag{5.9}$$

where $R_* \coloneqq r(X, \mathbf{x}^*) = r(\mathbf{x}^*, X)^\top$ is the $n \times 1$ matrix of correlations evaluated at all pairs of training and test points; and $R \coloneqq r(X, X) = (r(\mathbf{x}_i, \mathbf{x}_j))$ is the $n \times n$ correlation matrix of the training inputs.

This expression is useful to the link between regularization and Kriging in Section 5.4.2. In deriving the link, we shall pose attention to the predictive mean and variance of the Kriging emulator. Indeed, most of the machine learning literature focuses on expectation, because this has direct relevance for predictions. Here, we observe that a Nystöm regularization with negative variance would impair the use of the faster-Kriging for simulation optimization, which, instead, is relevant in operations research applications. We then pay attention to this aspect in deriving our emulator, so that no negative variance is achieved.

5.4.2 Linking regularization and Kriging

In this section, we present the core of the proposed methodology. The first intuition is to show that the regularization predictor can be obtained from applying ordinary Kriging with noisy observations (Williams and Rasmussen, 2006; Durrande et al., 2013). To illustrate, one starts assuming that the output y is observed with white noise, that is

$$y = g(\mathbf{x}) + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2), \tag{5.10}$$

and one adopts the ordinary Kriging meta-model without trend, rewriting Eq. (5.1) as

$$y = \mathcal{M}(\mathbf{x}), \qquad \mathcal{M}(\mathbf{x}) \sim \mathcal{GP}(0, \sigma^2 r(\mathbf{x}, \mathbf{x}')). \tag{5.11}$$

Thus, we have $cov(\mathbf{y}, \mathbf{y}) = cov(\underline{\mathcal{M}}, \underline{\mathcal{M}}) + \sigma_\epsilon^2 I_n = \sigma^2 R + \sigma_\epsilon^2 I_n$, where $\underline{\mathcal{M}} = (\mathcal{M}(\mathbf{x}_1), \ldots, \mathcal{M}(\mathbf{x}_n))^\top$.

Consider now making a prediction of the output value y^* given a new input value $\mathbf{x}^* \in \mathcal{X}$. By the properties of Gaussian processes, $\underline{\mathcal{M}}$ and $\mathcal{M}(\mathbf{x}^*)$ follow a multivariate Gaussian distribution:

$$\begin{pmatrix} \underline{\mathcal{M}} \\ \mathcal{M}(\mathbf{x}^*) \end{pmatrix} \Big| X, \mathbf{x}^*, \sigma^2, \sigma_\epsilon^2, \theta \sim \mathcal{N}\left[\mathbf{0}, \begin{pmatrix} \sigma^2 R + \sigma_\epsilon^2 I_n & \sigma^2 R_* \\ \sigma^2 R_*^\top & \sigma^2 \end{pmatrix} \right], \tag{5.12}$$

The Kriging predictor then follows the conditional distribution of $y^* := \mathcal{M}(\mathbf{x}^*)|\mathbf{y}, X, \mathbf{x}^*, \sigma^2, \sigma_\epsilon^2, \theta$, so that its mean and variance are given by

$$\begin{aligned} \mu_{y^*}(\mathbf{x}^*) &= \sigma^2 R_*^\top \left(\sigma^2 R + \sigma_\epsilon^2 I_n\right)^{-1} \mathbf{y} \\ &= R_*^\top \left(R + \frac{\sigma_\epsilon^2}{\sigma^2} I_n\right)^{-1} \mathbf{y} \qquad (5.13) \\ \sigma_{y^*}^2(\mathbf{x}^*) &= \sigma^2 - \sigma^2 R_*^\top \left(\sigma^2 R + \sigma_\epsilon^2 I_n\right)^{-1} \sigma^2 R_* \\ &= \sigma^2 \left(1 - R_*^\top \left(R + \frac{\sigma_\epsilon^2}{\sigma^2} I_n\right)^{-1} R_*\right). \qquad (5.14) \end{aligned}$$

Comparing Eq. (5.13) with Eq. (5.9), we see that the regularization predictor coincides with the Kriging predictor with noisy observations. In particular, the noise distribution is $\mathcal{N}(0, n\beta)$.

However, there are computational challenges associated with the Kriging method. In particular, the matrix R has size $n \times n$. Then, evaluating the inverse matrix R^{-1} becomes computationally impractical as n increases for reasons related to the speed with which the operations are performed and to the available memory. This problem has been addressed in the machine learning literature with alternative techniques, and a successful approach is Nyström regularization.

5.4.3 Nystöm regularization

The Nystöm method is used to approximate the covariance matrix K (Williams and Seeger, 2001; Williams and Rasmussen, 2006). To illustrate the idea of Nystöm regularization, we start with the expression of kernels regarding their eigenfunctions and eigenvalues using Mercer's theorem.

Let $\mathbb{P}$ be a measure over $\mathcal{X}$. A function $\varphi(\cdot)$ is called an eigenfunction of the kernel k with eigenvalue λ with respect to measure $\mathbb{P}$, if it satisfies the following integral:

$$\int k(\mathbf{x}, \mathbf{x}')\varphi(\mathbf{x})\mathrm{d}\mathbb{P}(\mathbf{x}) = \lambda\varphi(\mathbf{x}'). \tag{5.15}$$

Equation (5.15) possibly admits infinitely many solutions. We order the eigenfunctions as $\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \ldots$ according to their corresponding eigenvalues such that $\lambda_1 \geq \lambda_2, \ldots$. Eigenfunctions are orthogonal with respect to $\mathbb{P}$. Besides, the normalised eigenfunctions satisfy $\int \varphi_i(\mathbf{x})\varphi_j(\mathbf{x})\mathrm{d}\mathbb{P}(\mathbf{x}) = \delta_{ij}$, where δ_{ij} is the Kronecker delta.

Under certain regularity conditions, Mercer's theorem (König, 1986) allows us to express the kernel by normalized eigenfunctions and the corresponding eigenvalues:

$$k(\mathbf{x}, \mathbf{x}') = \sum_{l=1}^{\infty} \lambda_l \varphi_l(\mathbf{x})\varphi_l(\mathbf{x}'). \tag{5.16}$$

The right hand side in Eq. (5.16) can be a sum of infinitely many terms or can terminate at some value $t \in \mathbb{N}$ (in that case we write $\lambda_l = 0$ for $l > t$).

Given an i.i.d. sample $\{\mathbf{x}_j\}_{j=1}^n$ from a probability measure $\mathbb{P}_{\mathbf{X}}(\mathbf{x})$ [4], one considers the approximation:

$$\lambda_l\varphi_l(\mathbf{x}') = \int k(\mathbf{x}, \mathbf{x}')\varphi(\mathbf{x})\mathrm{d}\mathbb{P}_{\mathbf{X}}(\mathbf{x}) \approx \frac{1}{n}\sum_{j=1}^{n} k(\mathbf{x}_j, \mathbf{x}')\varphi_l(\mathbf{x}_j). \tag{5.17}$$

Letting $\mathbf{x}' = \mathbf{x}_j, j = 1 \ldots n$, one can write

$$\lambda_l\varphi_l(X) \approx \frac{1}{n}K\varphi_l(X), \tag{5.18}$$

where $\varphi_l(X) = (\varphi_l(\mathbf{x}_1), \ldots, \varphi_l(\mathbf{x}_n))^\top$, and K is the covariance matrix of sample $\{\mathbf{x}_j\}_{j=1}^n$. Considering the eigenproblem of a matrix K, one has

$$\lambda_l^{mat} \cdot \mathbf{u}_l = K\mathbf{u}_l, \tag{5.19}$$

where λ_l^{mat} is the l-th eigenvalue of a matrix K, and $\mathbf{u}_l$ is the corresponding $n \times 1$

[4] Here, we are interested in $\mathbb{P} = \mathbb{P}_{\mathbf{X}}$.

normalized eigenvector, $\mathbf{u}_l^\top \mathbf{u}_l = 1$. Comparing Eqs. (5.18) and (5.19), one obtains:

$$\lambda_l \approx \frac{1}{n}\lambda_l^{mat}, \qquad \varphi_l(\mathbf{x}_i) \approx \sqrt{n} u_{l,i}, \qquad i = 1 \dots n, \tag{5.20}$$

where $u_{l,i}$ represents the i-th entry of the eigenvector $\mathbf{u}_l$.[5] Plugging Eq. (5.20) back into Eq. (5.17), we obtain the Nystöm approximation to the l-th eigenfunction (Baker, 1977, Chapter 3):

$$\lambda_l \approx \frac{1}{n}\lambda_l^{mat}, \qquad \varphi_l(\mathbf{x}') \approx \frac{\sqrt{n}}{\lambda_l^{mat}} k(X, \mathbf{x}')^\top \mathbf{u}_l, \qquad \forall \mathbf{x}' \in \mathcal{X} \tag{5.21}$$

where $k(X, \mathbf{x}') = (k(\mathbf{x}_1, \mathbf{x}'), \dots, k(\mathbf{x}_n, \mathbf{x}'))^\top$. Note that Eq. (5.21) extends Eq. (5.20) from a sample $\{\mathbf{x}_j\}_{j=1}^n$ to the entire $\mathcal{X}$.

We now have all the tools to use the Nystöm method for approximating the covariance matrix K. Assume we select a subset of size m with $m < n$ (active set henceforth), without loss of generality, we assume the data points are ordered in such a way that the active set comes first, so that the corresponding design matrix is $X_m = (\mathbf{x}_1, \dots, \mathbf{x}_m)$. Then K is partitioned as:

$$K = \begin{bmatrix} K_{mm} & K_{m(n-m)} \\ K_{(n-m)m} & K_{(n-m)(n-m)} \end{bmatrix} \tag{5.22}$$

We also denote the upper block of size $m \times n$ as K_{mn}, $K_{mn} = [K_{mm}, K_{m(n-m)}]$ (by the symmetry of K, $K_{nm} = K_{mn}^\top$).

Let us compute the eigenvalues/vectors of the matrix K_{mm}, denoted with $\{\lambda_l^{(m)}\}_{l=1}^m$ and $\{\mathbf{u}_l^{(m)}\}_{l=1}^m$ henceforth. Applying Eq. (5.21), one has:

$$\tilde{\lambda}_l = \frac{\lambda_l^{(m)}}{m}, \qquad \tilde{\varphi}_l(\mathbf{x}') = \frac{\sqrt{m}}{\lambda_l^{(m)}} k(X_m, \mathbf{x}')^\top \mathbf{u}_l^{(m)}, \qquad \forall \mathbf{x}' \in \mathcal{X}, \tag{5.23}$$

where $k(X_m, \mathbf{x}') = (k(\mathbf{x}_1, \mathbf{x}'), \dots, k(\mathbf{x}_m, \mathbf{x}'))^\top$ [6].

Applying Mercer's theorem (Eq. (5.16)), and truncating at $t = m$, we have the Nystöm approximation for a kernel using a subset of size m of the training points:

[5] For a fixed l, $1/n\lambda_l^{mat} \overset{n\to\infty}{\longrightarrow} \lambda_l$

[6] $\tilde{\cdot}$ is used for indicating an approximation.

$$\begin{aligned}\tilde{k}(\mathbf{x},\mathbf{x}') &= \sum_{l=1}^{m} \frac{\lambda_l^{(m)}}{m} \tilde{\varphi}_l(\mathbf{x})\tilde{\varphi}_l(\mathbf{x}') \\ &= \sum_{l=1}^{m} \frac{\lambda_l^{(m)}}{m} \left(\frac{\sqrt{m}}{\lambda_l^{(m)}}\right)^2 k(X_m,\mathbf{x})^\top \mathbf{u}_l^{(m)} \cdot \left(\mathbf{u}_l^{(m)}\right)^\top k(X_m,\mathbf{x}') \\ &= k(X_m,\mathbf{x})^\top K_{mm}^{-1} k(X_m,\mathbf{x}'). \end{aligned} \tag{5.24}$$

Now evaluating Eq. (5.24) for all pairs of $\{\mathbf{x}_j\}_{j=1}^n$ and the new point $\mathbf{x}^*$, we obtain the following equations:

$$\tilde{K} = K_{nm} K_{mm}^{-1} K_{mn}, \tag{5.25}$$

$$\tilde{k}(X,\mathbf{x}^*) = K_{nm} K_{mm}^{-1} k(X_m,\mathbf{x}^*). \tag{5.26}$$

Plugging Eqs. (5.25) and (5.26) into the regularization predictor in Eq. (5.8), we obtain the Nyström-based predictor:

$$\tilde{y}^* = k(X_m,\mathbf{x}^*)^\top \left(K_{mn}K_{nm} + n\beta K_{mm}\right)^{-1} K_{mn}\mathbf{y} \tag{5.27}$$

$$= R_{mn^*}^\top \left(R_{mn}R_{nm} + \frac{n\beta}{\sigma^2} R_{mm}\right)^{-1} R_{mn}\mathbf{y}, \tag{5.28}$$

where $R_{mn^*} \coloneqq r(X_m,\mathbf{x}^*) = R_{n^*m}^\top$ and $R_{mm} \coloneqq r(X_m,X_m)$.

The above predictor Eq. (5.27) can be interpreted from a Kernel Regularized Least Squares viewpoint. Instead of considering the space $\mathcal{H}_n$ in Eq. (5.4), we seek the minimiser of the regularization problem in Eq. (5.6) in the following reduced space:

$$\mathcal{H}_m = \left\{ f \in \mathcal{H} | f(\mathbf{x}) = \sum_{j=1}^{m} \tilde{\alpha}_j k(\mathbf{x}_j,\mathbf{x}), \mathbf{x}_j \in \mathcal{X}, \tilde{\alpha}_j \in \mathbb{R} \right\} \tag{5.29}$$

where $\{\mathbf{x}_j\}_{j=1}^m$ is a subset of the input training set. The corresponding solution to Eq. 5.6 now becomes:

$$\widehat{f}_{\beta,m}(\cdot) = \sum_{j=1}^{m} \tilde{\alpha}_j k(\mathbf{x}_j,\cdot) \quad \text{with} \quad \tilde{\alpha} = \left(K_{mn}K_{nm} + n\beta K_{mm}\right)^{-1} K_{mn}\mathbf{y}. \tag{5.30}$$

By the equivalence between the regularization predictor and Kriging predictor, i.e. Eq. (5.9) and Eq. (5.13), we can derive the Nyström predictive variance by plugging Eqs.

(5.25) and (5.26) into Eq. (5.14), and obtain

$$\sigma^2_{\tilde{y}^*} = \sigma^2 \left(1 - \left(R_{mn}^\top R_{mm}^{-1} R_{mn^*} \right)^\top \left(R_{nm} R_{mm}^{-1} R_{mn} + \frac{n\beta}{\sigma^2} I_n \right)^{-1} \left(R_{mn}^\top R_{mm}^{-1} R_{mn^*} \right) \right). \quad (5.31)$$

Recall that β is not a random variable with distribution but, rather, a choice regarding the trade-off between fit and complexity made by the analyst. The Nystöm approximation is, in fact, a reduced-rank approximation, for this approximation to be accurate, we are hoping the kernel to be a fast-decaying eigenspectrum (or faster decay of eigenvalues).

5.4.4 Parameter estimation

In the Nystöm approximated Kriging predictor (Eqs. (5.28) and (5.31)), unknown parameters include the kernel bandwidth θ and the regularization parameter β, denoted by $\psi = (\theta, \beta)$. We estimate ψ by minimizing the expected squared risk:

$$\psi = \underset{\psi}{\operatorname{argmin}} \int \left(f_{\beta,m}(\mathbf{x}) - y \right)^2 \mathrm{d}\mathbb{P}(\mathbf{x}, y), \quad (5.32)$$

where $\mathbb{P}$ denotes the distribution associated with the input-output probability space. Based on the training dataset, the numerical solution of above equation writes:

$$\widehat{\psi} \approx \underset{\psi}{\operatorname{argmin}} \frac{1}{n} \sum_{j=1}^{n} \left(f_{\beta,m}(\mathbf{x}_j) - y_j \right)^2. \quad (5.33)$$

Besides, the variance of Gaussian process σ^2 in Eq. 5.11 is estimated by plugging $\widehat{\psi}$ into

$$\widehat{\sigma}^2 = \frac{1}{n} \mathbf{y}^\top \tilde{R} \mathbf{y} = \frac{1}{n} \mathbf{y}^\top R_{nm} R_{mm}^{-1} R_{mn} \mathbf{y}^\top.$$

5.4.5 Computational analysis of the algorithm

A practical implementation of the Nystöm approximated Kriging predictor is shown in Algorithm 1. This algorithm has been coded in MATLAB, and we call this emulator 'fast Kriging'.

The steps $1, 2, 4$ of the presented algorithm require $O(nmd)$ in time and $O(nm)$ in memory to compute the R_{nm}, $O(m^3 + m^2 d)$ in time and $O(m^2)$ in memory to compute R_{mm} and perform the Cholesky decomposition. Moreover $O(nm^2 + m^3 + nm)$ in time and

Algorithm 1: Fast Kriging algorithm

Input: X, $\mathbf{y}$, $r(\cdot)$, $\mathbf{x}^*$, m

1. Randomly sample m columns from X and obtain X_m;
   Randomly split X into $[X_t, X_v]$, where X_t has size $d \times n_t$ with $n_t = 0.3n$; and X_v has column size $n_v = n - n_t$; then obtain corresponding $\mathbf{y} = (\mathbf{y}_t, \mathbf{y}_v)^\top$, where $\mathbf{y}_v = (y_{v,1}, \ldots, y_{v,n_v})^\top$.

2. Define the following functions of $\psi = (\gamma, \theta)$:

$$L_0 := chol(R_{mm}); A_0 := R_{n_t m} L_0^{-1}; R_0 := (A_0^\top A_0 + n_t \gamma I_m); \alpha_0 := L_0^{-1} R_0^{-1} A_0^\top \mathbf{y}_t;$$

   and

$$\tilde{\mathbf{y}}_v := R_{n_v m} \alpha,$$

   where

$$R_{n_t m} := r(X_t, X_m); R_{n_v m} := r(X_v, X_m); \tilde{\mathbf{y}}_v = (\tilde{y}_{v,1}, \ldots, \tilde{y}_{v,n_v})^\top.$$

3. Let

$$\widehat{\psi} = (\widehat{\gamma}, \widehat{\theta}) = \underset{\psi}{\operatorname{argmin}} \frac{1}{n_v} \sum_{j=1}^{n_v} \left(\tilde{y}_{v,j} - y_{v,j}\right)^2.$$

4. Compute R_{mm}, R_{nm} and R_{mn^*} using $\widehat{\theta}$, then calculate

$$L_1 := chol(R_{mm}); A_1 := R_{nm} L_1^{-1}; R_1 := (A_1^\top A_1 + n\widehat{\gamma} I_m); \alpha_1 := L_1^{-1} R_1^{-1} A_1^\top \mathbf{y};$$

   and

$$\tilde{y}^* = R_{mn^*}^\top \cdot \alpha_1.$$

   The prediction variance is obtained by:

$$\sigma^2_{\tilde{y}^*} := \widehat{\sigma}^2 \left(1 - diag\{R_{mn^*}^\top W R_{mn^*}\}\right),$$

   where

$$\begin{aligned} \widehat{\sigma}^2 &:= \frac{1}{n} Z^\top Z, \quad \text{with} \quad Z := A_1^{-1} \mathbf{y}; \\ W &:= (L_1)^{-1} R_1^{-1} A_1^\top \cdot A_1 (L_1^\top)^{-1}. \end{aligned}$$

Output: $\tilde{y}^*, \widehat{\sigma}^2_{\tilde{y}^*}$

[a] $\gamma = \frac{\beta}{\sigma^2}$ in Eqs. (5.28) and (5.31)

[b] $L = chol(B)$: Cholesky decomposition of B; L is a lower triangular matrix such that $LL^\top = B$

$O(m^2)$ in memory to compute $A_0^\top A_0$ and α_0. For a total cost of

$$O(nm(d+m)) \text{ in time}, \quad O(m^2+n) \text{ in space}.$$

The step 3 of the algorithm is performed via stochastic coordinate descent (Nesterov, 2012) and requires the cost of steps $1, 2$ times the number of parameters in ψ and the maximum number of descent steps T allowed, for a total time cost of

$$O(nm(d+m)|\psi|T).$$

Note that for stochastic coordinate descent the error on the solution at step T with respect to the solution at the local minimum is in the order of $O(\frac{1}{\sqrt{T}})$.

Finally, the cost of the algorithm in the test/prediction phase (step 4) is $O(n^*m)$ in time to compute the mean values for the prediction set and $O(n^*m^2)$ in time for the variance, where n^* is the number of prediction points. Note that the computation of W is done once in the training phase and costs $O(nm^2)$. So in conclusion

$$\begin{aligned} Train: &\quad O(nm(d+m)|\psi|T) \text{ in time}, \quad O(m^2+n) \text{ in space} \\ Test: &\quad O(n^*m^2) \text{ in time}, \quad O(m^2+n^*) \text{ in space}. \end{aligned}$$

5.4.6 Theoretical analysis

In this subsection we quantify the point-wise discrepancy between the standard Kriging estimator and the proposed approximation. We use the notation introduced in Section 5.4.2. Rudi et al. (2015) analyze the generalization properties of the Nyström approximation method in the context of statistical machine learning, where the data are assumed to be independently and identically distributed. Here, by using similar techniques, we analyze the properties of the Nyström approximation for Gaussian processes.

Theorem 5.4.1. *Let $\delta \in (0,1]$, $m, n \in \mathbb{N}$ with $m \leq n$ and $\gamma > 0$. Assume that there exists $\kappa > 0$ such that $k(\mathbf{x},\mathbf{x}) \leq \kappa^2$ for any $\mathbf{x} \in \mathcal{X}$. Assume moreover that the Nyström points are selected uniformly at random from the dataset. Denote with $\widehat{f}$ the Kriging function and with $\widetilde{f}$ the Nyström Kriging function in Algorithm 1 and with α the Kriging coefficients $\alpha = (R+\gamma nI)^{-1}\mathbf{y}$. Then, for any $\mathbf{x}_* \in \mathcal{X}$, we have*

$$|\widehat{f}(\mathbf{x}_*) - \widetilde{f}(\mathbf{x}_*)| \leq \sqrt{\frac{30\kappa^4\|\alpha\| \log\frac{40\kappa^2}{m\delta}}{m}} + \frac{30\kappa^3\|\alpha\| \log\frac{40\kappa^2}{m\delta}}{m\sqrt{\gamma}}.$$

Proof. 5.4.1. Let $z_1, \ldots, z_n \in \mathcal{H}$ and for all $f \in \mathcal{H}, \alpha \in \mathbb{R}^m$, define as (see appendix of

Rudi et al. (2015))

$$S : \mathcal{H} \to \mathbb{R}^n, \qquad S(f) = \frac{1}{\sqrt{n}}(\langle z_1, f\rangle_{\mathcal{H}}, \ldots, \langle z_n, f\rangle_{\mathcal{H}})$$
$$S^* : \mathbb{R}^n \to \mathcal{H}, \qquad S^*(\alpha) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n} \alpha_i z_i.$$

Note in particular that $SS^* = R_{nn}/n$ (see appendix of (Rudi et al., 2015)). Denote with P the orthogonal projection operator whose range is the subspace $\mathcal{H}_m = \text{span}\{k(\cdot, \widetilde{\mathbf{x}}_1), \ldots, k(\cdot, \widetilde{\mathbf{x}}_m)\}$, where $\{\widetilde{\mathbf{x}}_j\}_{j=1}^m$ are the selected Nyström points. In Caponnetto and De Vito (2007), it is shown that the Kriging mean function $\widehat{f} \in \mathcal{H}$ is characterized by

$$\widehat{f} = S^*(SS^* + \gamma I)^{-1}\widehat{y}, \quad \widehat{y} = \frac{1}{\sqrt{n}}(y_1, \ldots, y_n).$$

While, by Lemma 2 of Rudi et al. (2015) and the spectral theorem, we have that the Nyström approximation $\widetilde{f} \in \mathcal{H}$ of the Kriging estimator is characterized by

$$\widetilde{f} = PS^*(SPS^* + \gamma I)^{-1}\widehat{y}.$$

Let $\mathbf{x}_*$ be a test point. We have

$$\begin{aligned}\widetilde{f}(\mathbf{x}_*) - \widehat{f}(\mathbf{x}_*) &= \left\langle k(\cdot, x_i), PS^*(SPS^* + \gamma I)^{-1}\widehat{y} - S^*(SS^* + \gamma I)^{-1}\widehat{y}\right\rangle \\ &= \left\langle k(\cdot, \mathbf{x}_i), (P - I)S^*(SS^* + \gamma I)^{-1}\widehat{y}\right\rangle \\ &\quad + \left\langle k(\cdot, \mathbf{x}_i), PS^*[(SPS^* + \gamma I)^{-1} - (SS^* + \gamma I)^{-1}]\widehat{y}\right\rangle.\end{aligned}$$

Now for the first term, we have

$$|\left\langle k(\cdot, \mathbf{x}_i), (P - I)S^*(SS^* + \gamma I)^{-1}\widehat{y}\right\rangle| \leq \|k(\cdot, \mathbf{x}_i)\|_{\mathcal{H}}\|(I - P)S^*\|\|(SS^* + \gamma I)^{-1}\widehat{y}\|.$$

Note in particular that $\|k(\cdot, \mathbf{x}_i)\| = \sqrt{k(\mathbf{x}_i, \mathbf{x}_i)} \leq \kappa$. Moreover $(SS^* + \gamma I)^{-1}\widehat{y} = \alpha$.

For the second term note that by the matrix identity $A^{-1} - B^{-1} = A^{-1}(B - A)B^{-1}$ valid for any two invertible matrices, we have

$$\begin{aligned}|\left\langle k(\cdot, \mathbf{x}_i), PS^*[(SPS^* + \gamma I)^{-1} - (SS^* + \gamma I)^{-1}]\widehat{y}\right\rangle| &= \\ |\left\langle k(\cdot, \mathbf{x}_i), PS^*(SPS^* + \gamma I)^{-1}(SS^* - SPS^*)(SS^* + \gamma I)^{-1}\widehat{y}\right\rangle| &\\ \leq \|k(\cdot, \mathbf{x}_i)\|_{\mathcal{H}}\|PS^*(SPS^* + \gamma I)^{-1/2}\| \times &\\ \times \|(SPS^* + \gamma I)^{-1/2}\|\ \|SS^* - SPS^*\|\ \|(SS^* + \gamma I)^{-1}\widehat{y}\|.&\end{aligned}$$

In particular we have $\|(SPS^* + \gamma I)^{-1/2}\| \leq \gamma^{-1}$, moreover by the fact that $\|Z\|^2 = \|Z^*Z\|$

for any bounded operator and that $P^2 = P$ since it is a projection operator, we have

$$\|PS^*(SPS^* + \gamma I)^{-1/2}\|^2 = \|(SPS^* + \gamma I)^{-1/2} SPS^* (SPS^* + \gamma I)^{-1/2}\| \leq 1,$$

moreover, since $I - P$ is a projection operator, then $(I - P) = (I - P)^2$, so

$$\|SS^* - SPS^*\| = \|S(I - P)S^*\| = \|S(I - P)^2 S^*\| = \|(I - P)S^*\|^2.$$

In particular, since $S^*S + \eta I - S^*S$ is positive semidefinite for any η, by Prop. 5 of Rudi et al. (2015), we have that

$$\|(I - P)S^*\|^2 \leq \|(I - P)(S^*S + \eta I)^{1/2}\|^2.$$

Finally, by Lemma 6 of Rudi et al. (2015), by selecting $\eta = \frac{10\kappa^2}{m} \log \frac{40\kappa^2}{m\delta}$, we have

$$\|(I - P)S^*\|^2 \leq 3\eta,$$

with probability $1 - \delta$. So finally

$$|\widehat{f}(\mathbf{x}_*) - \widetilde{f}(\mathbf{x}_*)| \leq \kappa\sqrt{3\eta}\|\alpha\| + \frac{3\eta\kappa\|\alpha\|}{\sqrt{\gamma}}.$$

□

Theorem 5.4.1 provides a guarantee on the accuracy of the algorithm. It also reveals that the choice of m is important in a practical implementation of the algorithm. Rudi et al. (2015) provide a strategy to efficiently explore different subsampling levels m. The idea is to use rank-one Cholesky updates in steps 2 and 4 of Algorithm 1. 5.6 provides a detailed discussion.

5.5 Numerical experiments

The experiments are divided into two stages. In the first stage, detailed in this section, we study the performance of the proposed fast Kriging in terms of fitting, speed and memory requirements. In the second stage (next section), we test the emulator in the challenging numerical task of estimating global sensitivity measures.

We compare the performance of fast Kriging with four standard Kriging subroutines. The first is the UQlab Kriging subroutine, which is a MATLAB meta-modeling tool created by Lataniotis et al. (2015). The second is a well-known MATLAB toolbox called 'DACE', developed by Lophaven et al. (2002). The third and fourth Kriging subroutines are functions belonging to the R package `laGP`, which is developed by Gramacy (2016).

Specifically, we consider the standard Kriging emulator `GP`, and the Kriging emulator for large dataset called `aGP`. The heart of the `laGP` package is a C implementation; thus, it should be faster than a basic R implementation. All above Kriging subroutines are open-source and free to download.

To assess fitting accuracy, we compare the actual response surface and the predicted values at a large number of prediction points. We use two diagnostic statistics to assess the prediction performance: the root mean squared errors and the coefficient of determination (R^2). Assume we have n_{pred} new prediction points $\{\mathbf{x}_j^*\}_{j=1}^{n_{pred}}$, and corresponding true output values and fitted values are denoted with $\{y_j^*\}_{j=1}^{n_{pred}}$ and $\{\hat{y}_j\}_{j=1}^{n_{pred}}$ respectively. The two diagnostic statistics are estimated empirically as follows [7]:

$$\text{RMSE} \approx \sqrt{\frac{1}{n_{pred}} \sum_{j=1}^{n_{pred}} (\hat{y}_j - y_j^*)^2}, \quad \text{R}^2 \approx 1 - \frac{\sum_{j=1}^{n_{pred}} (\hat{y}_j - y_j^*)^2}{\sum_{j=1}^{n_{pred}} (y_j^* - \bar{y}^*)^2}, \tag{5.34}$$

where $\bar{y}^*$ is the mean of $\{y_j^*\}_{j=1}^{n_{pred}}$.
To evaluate the speed of training and prediction, we record the elapsed time for training and the elapsed time for a new prediction (average over n_{pred} predictions). For the `aGP` algorithm, the training and prediction time are bound together. Since the main use of emulators is to replace the simulator and provide predictions for analysis, we focus on the prediction and record the average prediction time of `aGP` as the mean of the total execution time (training and prediction).

To assess the capability of analyzing large datasets, we record the maximum training set one can use on a standard PC. We conduct the numerical experiments on a PC with an Intel-i5 core processor at 3.10GHz and 4G RAM. The software we are using are MATLAB 2016a, R 3.4.1.

5.5.1 21-input simulator

We first test the Kriging subroutines on the 21-input simulator, i.e. the additive Gaussian function in Eq. (4.16).

We generate the training and prediction inputs by Quasi-Monte Carlo. We set the correlation function as the Gaussian kernel $\exp(-\frac{h^2}{2\theta})$ with zero trend ($\phi(\mathbf{x})^\top \mathbf{w} = 0$) for all five subroutines, so that the number of unknown Kriging parameters are the same. We conduct 30 replicates of the experiments. For fast Kriging, we choose the subsample size $m = 100$, following the selection method in Section 5.6. For this simple additive model (4.16), we are expecting accurate approximations from all Kriging subroutines. Table 5.2

[7] In order to illustrate results with a clear notation, we start using n_{train} and n_{pred} for number of training and prediction cases.

Table 5.2: Meta-model performance for the 21-input simulator in Eq. (4.16), $n_{train} = 1000$.

	T_{train} (sec)	T_{pred} (10^{-6}sec)	Average RMSE	Mean R^2	Max n_{train}
Fast Kriging ($m = 100$)	0.12 ± 0.05	2.44 ± 0.12	0.05	0.99	$> 40,000$
UQLab	43.59 ± 4.04	93 ± 2.86	0.40	0.99	7,000
DACE	43.76 ± 0.34	956.97 ± 13.35	0.01	0.99	2,300
`laGP.GP`	18.39 ± 0.16	917.88 ± 7.50	0.86	0.99	10,000
`laGP.aGP`	NA	5.45×10^4	0.79	0.99	10,000

illustrates the Kriging subroutines performance for model (4.16). Our goal is to test the training and prediction time when all subroutines fit well the response surface. For this simple model, a training set of $n_{train} = 1000$ is sufficient for an accurate approximation: columns four and five in Table 5.2 display coefficients of determination close to unity and small RMSE's for all subroutines. However, note that to use the DACE subroutine, one must provide the upper and lower bounds for θ, and the fitting performance depends strongly on the choice of the bounds. We conducted several pre-training tests to obtain a proper choice.

Let us now come to training and prediction times (T_{train}, T_{pred}, henceforth), calculated based on the training set discussed above. The second column of Table 5.2 presents the mean and standard deviation of T_{train} over 30 replicates, results are in seconds. The replicates take into account that the optimization is associated with a random subsample coming from regularization and also with variability in the initialization seed of the optimization subroutine. The third column reports the average time for a new prediction over $n_{pred} = 3000$ predictions (results are in 10^{-6} seconds). Column 2 shows that fast kriging saves from 95% to 99.7% of the training time. In prediction, all subroutines show an average value of T_{pred} lower than 10^{-3} seconds. For prediction, fast Kriging saves 95% of time comparing to UQlab. Note that the total execution time (training on 1000 points and predicting on 3000 values) of `aGP` is 163 seconds, while fast Kriging takes less than 1 second.

The last column of Table 5.2 reports the maximum number of training points one can use on a PC with 4G RAM (exceeding this number causes an out-of-memory problem). With DACE, UQlab and `laGP`, the maximum sample size for training is $2,300$, $7,000$, and $10,000$ points respectively. With fast Kriging, the size can be higher than $n_{train} = 40,000$.

Thus, while providing similar results in terms of accuracy, the fast Kriging algorithm promises to be less demanding in terms of space and time than the other subroutines. We test this assertion further with two realistic-size simulators.

Table 5.3: Meta-model performance for LevelE code, $n_{train} = 1000$

	T_{train} (sec)	T_{pred} (10^-6 sec)	Average RMSE	Average R^2	Max n_{train}
Fast Kriging ($m = 400$)	1.17 ± 0.55	14.24 ± 0.33	0.0019	0.89	160,000
UQlab	38.75 ± 15.66	190.13 ± 49.89	0.0016	0.93	6,500
DACE	19.52 ± 0.04	791.74 ± 6.70	0.0020	0.89	3,500
`laGP.GP`	9.23 ± 0.05	1112.88 ± 8.82	0.0027	0.82	8,000
`laGP.aGP`	NA	4.85×10^4	0.0030	0.80	8,000

5.5.2 LevelE simulator

In this section, we apply the subroutines to the benchmark model in sensitivity analysis: LevelE, see Section 4.6 of Chapter 4 for further details.

In this section, we try to leave most options of the subroutines at the settings suggested by the respective manuals, because that is what most analysts are likely to do. The training and prediction sets have sizes of $n_{train} = 1000$ and $n_{pred} = 3000$ respectively. For the fast Kriging algorithm we set $m = 400$.

Table 5.3 shows the results for the LevelE code. One can see that, given 1000 training points, most of the subroutines perform satisfactorily with $R^2 > 82\%$. Regarding the training time to fit the surface, fast Kriging takes 1.17 seconds, while UQlab, DACE and `GP` take more than 9 seconds. In prediction, the mean single prediction time is of the order of 10^{-3} seconds or less, just slightly higher than for the 21-input simulator (fast kriging saves at least 64%). In terms of time per prediction, fast Kriging takes less than 13% of the time required by other subroutines. In terms of memory requirements and therefore the size of the largest allowable dataset, the DACE subroutine works without an out-of-memory result for $n_{train} \leq 3,500$, UQlab for $n_{train} \leq 6,500$, and `laGP` for $n_{train} \leq 8,000$. The fast Kriging subroutine did not produce an out-of-memory till $n_{train} = 160,000$.

5.5.3 STOCFOR3

In this section, we challenge the subroutines with an input-output dataset generated by a dimensionally large simulator: STOCFOR3. STOCFOR3 is the linear program (LP) of largest size available in the NETLIB Library. The problem parameters can be downloaded from the NETLIB website. The LP has then been implemented in MATLAB by the authors and solved through the `lingprog.m` subroutine with the dual-simplex method. As typical in the sensitivity analysis of linear programs (Wendell, 2004), we have considered a hyper-box for parameter variations, with coefficients and right-hand side (RHS) terms varying between $[-100\%. + 200\%]$ and $[\pm 100\%]$, respectively. This leads to a total of $40,216$ inputs ($23,541$ coefficients and $16,675$ RHS terms).

We evaluate the simulator on $40,000$ input points for obtaining an uncertainty quantification. We use the `rand.m` subroutine in MATLAB to generate the random model input sample, because the quasi-random generation algorithms of Sobol' and Halton are limited at $d = 1,111$. The number of training points is $n_{train} = 20,000$, while $n_{pred} = 10,000$. The size of the training dataset ($d \times n_{train} = 40,216 \times 20,000$) does not make the analysis feasible on a personal PC. We then use cloud computing services with the same memory specifications for all subroutines (up to 1 Tera Byte RAM). Only fast Kriging manages to produce results, the other subroutines fail due to memory requirements. In terms of accuracy and time, using $m = 12,000$, we register $\mathrm{R}^2 = 0.9376$, and the root relative squared error (RRSE) is 0.2498. [8] The training and prediction times are $T_{train} = 4.27$ hours and $T_{pred} = 0.0016$ seconds respectively.

5.6 Details on efficiently tuning the number of Nyström centers

According to Theorem 5.4.1, the efficiency of the Nyström approximation depends on the redundancy of the dataset and the coreset size (number of Nyström centers). In practice, to identify the number of Nyström centers (m), the analyst can adopt a sequential tuning procedure. Intuitively, one selects an increasing and finite sequence of values for m and stops at the value that satisfies certain criteria. The criteria usually depend on the prediction performance and on the gain in efficiency. As m increases, the accuracy of the approximation increases; however a high value of m leads to a poor gain in efficiency.

A typical choice for a stopping rule is the marginal improvement of a given performance measure achieved by adding points to the coreset. Here we provide three stopping rules based on the RMSE. Considering a window of size S on the last recorded RMSE's, one can stop adding points to the subset when one of the following conditions is met: 1) the relative decrease of the first and last RMSE's of the window is lower than a given threshold, i.e. $\frac{\mathrm{RMSE}_1 - \mathrm{RMSE}_S}{\mathrm{RMSE}_1} < \epsilon$; 2) the relative decrease of the average over the first and last 10% RMSE's of the window is lower than a given threshold, i.e. $\frac{\overline{\mathrm{RMSE}_{1:0.1S}} - \overline{\mathrm{RMSE}_{0.9S:S}}}{\overline{\mathrm{RMSE}_{1:0.1S}}} < \epsilon$; 3) similar to $\mathcal{C}_2$ but with the median replacing the mean.

This naive algorithm would correspond to a computational cost of $O(nm^2T + m^3T)$, where T is the number of candidate m's before the stopping criterion is met. However, Rudi et al. (2015, Algorithm 1) recommend an efficient incremental updating algorithm for tuning m. The main idea is the following: starting from a small value of m, increase

[8] In this case we cannot compare across subroutines. Therefore, we report the RRSE, which ranges from 0 to infinity, with 0 corresponding to the ideal. RRSE $\approx \sqrt{\frac{\sum(\hat{y}_j - y_j^*)^2}{\sum(y_j^* - \bar{y}^*)^2}}$.

m by one at each iteration. At iteration t, the matrix L_t in **Algorithm 1** is computed via the Cholesky rank-one update formula using L_{t-1} from the previous iteration. In this way, the computational cost of tuning m is reduced to $O(nm^2 + m^3)$. Figure 5.1 shows

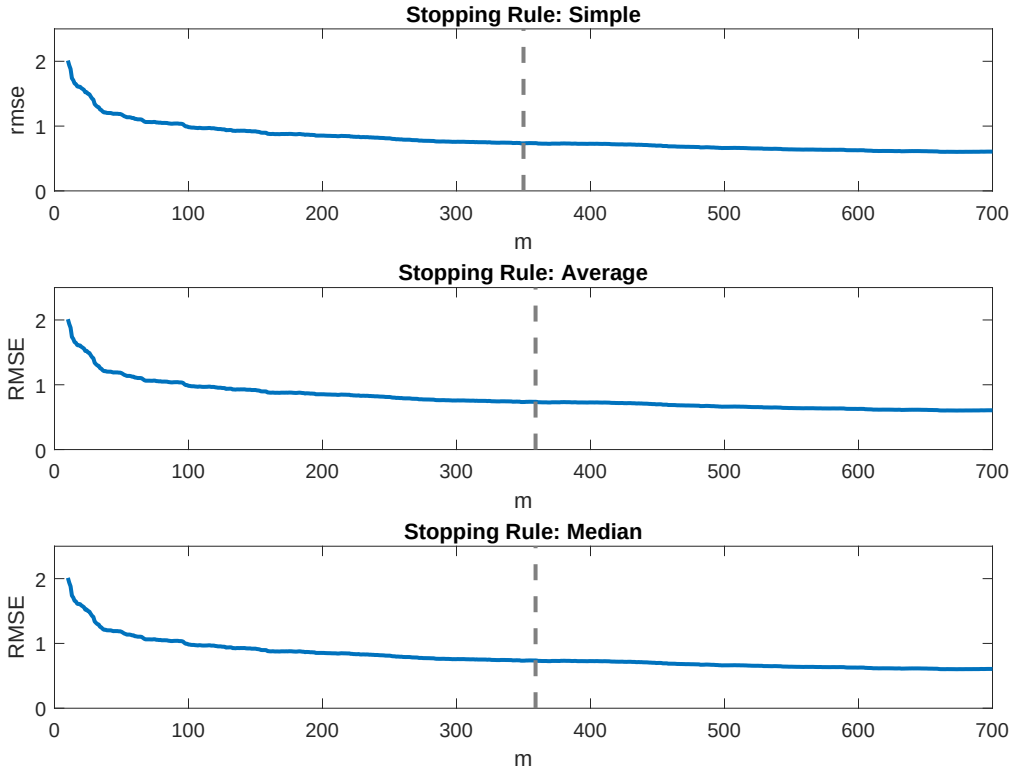


Figure 5.1: Results of stopping criteria for LevelE model with dataset of size $n = 1000$. Window size uses $S = 50$. Threshold is $\epsilon = 1\%$. Simple stopping rule suggests $m = 350$; Average and median stopping rules suggest $m = 359$.

an example of choosing m using the stopping criteria mentioned above, for the LevelE simulator. We use $n_{\text{train}} = 1000$ and increase m from 10 to 700. The threshold is set at $\epsilon = 1\%$. The first stopping rule suggests $m = 350$, while with the second and third rules one would obtain $m = 359$.

5.7 Application: estimating functionals of the output distribution

In this section, we challenge the proposed fast Kriging algorithm in the estimation of global sensitivity measures. The details on definitions and estimations of global sensitivity measures are presented in Sections 2.2.3, 2.2.5 of Chapter 2 and Section 4.2.1 of Chapter 4. We observe that the expressions of global sensitivity measures involve the determination of functionals of marginal and conditional the model output distributions,

$\mathbb{P}_Y$ and $\mathbb{P}_{Y|X_i}$. Then, if the emulator grants an accurate estimation of these functionals, the emulator is capable of correctly identifying the conditional and unconditional model output distributions, not only one of the moments of the simulator output. However, the estimation of these distributions requires a large number of emulator evaluations and speed becomes a central aspect. Here, our numerical experiments address the following research question. Given that the analyst replaces the original model with any of the four Kriging subroutines, once predictions have been generated, how accurate are the estimates and how much time is the estimation requiring?

Table 5.4: Global sensitivity measures estimates for the 21-input simulator in Eq. (4.16)

Sensitivity measure	η_i	δ_i	β_i^{KS}
Analytical values			
$X^1 \dots X^7$	0.108	0.112	0.110
$X^8 \dots X^{14}$	0.027	0.053	0.053
$X^{15} \dots X^{21}$	0.006	0.026	0.026
Fast Kriging: running time =16.90 s			
$X^1 \dots X^7$	0.115	0.114	0.112
$X^8 \dots X^{14}$	0.027	0.053	0.053
$X^{15} \dots X^{21}$	0.006	0.026	0.026
UQLab: runining time = 304.25 s			
$X^1 \dots X^7$	0.108	0.109	0.106
$X^8 \dots X^{14}$	0.025	0.051	0.052
$X^{15} \dots X^{21}$	0.005	0.025	0.025

Estimates for the 21-input simulator in Eq. (4.16), LevelE code and STOCFOR3 are discussed. We restrict the comparison to the fastest subroutines, namely fast Kriging and UQLab. In fact, as we have seen before, the execution time `aGP` is notably slower than that of fast Kriging and UQLab and in the prediction on massive new values the `aGP` execution time becomes extremely long.

We start with the 21-input simulator. The emulators are trained with $n_{train} = 1,000$. A brute force approach here requires $d \cdot n_{ext} \cdot n_{int}$ evaluations. Because $d = 21$ and we set $n_{ext} = n_{int} = 1,000$, a total of $21,000,000$ evaluations (predictions) of the emulator is required. For this simulator, we have the analytical values for the global sensitivity measures (lines 3 to 5 in Table 5.4) and it is therefore possible to compare the analytical values with the estimates.

The simulator inputs are split into three groups of equal importance: $X^1 \dots X^7$, followed by $X^8 \dots X^{14}$, and by $X^{15} \dots X^{21}$. fast Kriging and UQlab clearly provide accurate estimates. However, fast Kriging requires 16.90 seconds and UQlab 304.25 seconds.

We now come to the LevelE code. For this code, analytical expressions of the global

sensitivity measures are not available. However, the simulator has been intensively studied in previous sensitivity analysis works (Saltelli et al., 2000; Borgonovo et al., 2012; Plischke and Borgonovo, 2017), and X^4 and X^{12} have been identified as the most important inputs. For more details on sensitivity analysis of LevelE code, see Section 4.6 of Chapter 4.

Our computational test is then as follows. The fast Kriging and UQlab emulators are trained at increasing sample sizes, from $n_{train} = 100$ to $n_{train} = 1000$. For each training set, we generate predictions of size $n_{pred} = 162,144$ from which we estimate the global sensitivity measures using subroutines based on a given-data approach.

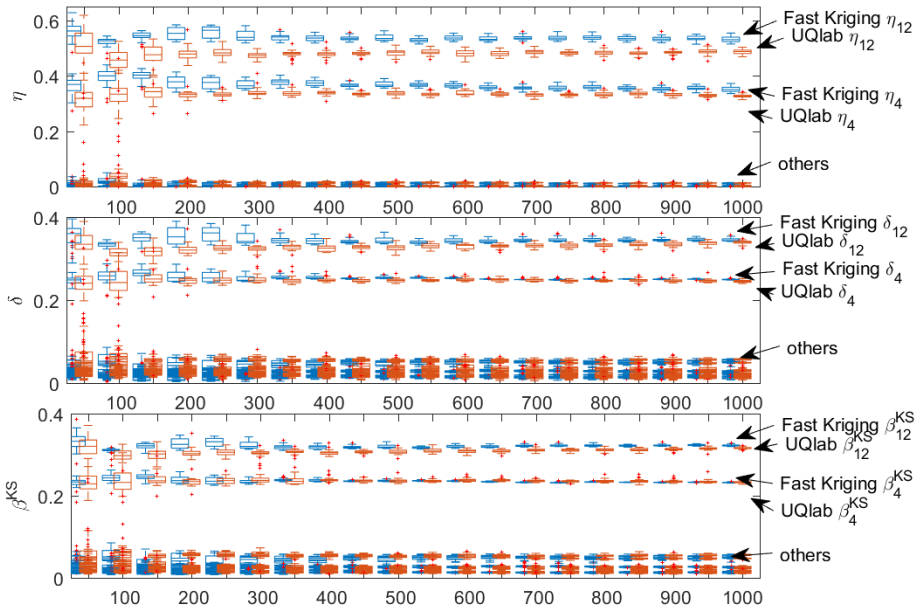


Figure 5.2: Global sensitivity measure estimates for LevelE code using fast Kriging and UQLab Kriging subroutines. Three sensitivity measures are considered: η_i, δ_i and β_i^{KS}. In each plot, the X-axis indicates the training size increasing from 100 to 1000. The blue boxes stand for the fast Kriging estimates, the red for UQlab. For LevelE model, two out of 12 inputs are the key drivers. Both Kriging and UQlab estimates clearly identify the rank of the inputs.

Figure 5.2 displays the sensitivity measures estimates, obtained using fast Kriging and UQLab as n_{train} increases. The error bands are produced using 30 replicates. We observe convergence as n_{train} increases, the error bands collapsing towards point estimates. Notably, the identification of the two most important inputs is achieved clearly already at $n_{train} = 400$. This would imply that a computational strategy requiring a number of model evaluations below $1,000$ would lead to the correct identification of the key uncertainty drivers. Note that this number is in line with current best practices in the literature.

Regarding the time for analysis, results are reported in Figure 5.3. The execution time of UQLab increases as training size increases: at $n_{train} = 400$, UQLab takes $T_{est} = 20$ seconds on average for estimation, while at $n_{train} = 1000$, the average estimation time escalate to $T_{est} = 304$ seconds. The required estimation time of fast Kriging remains at

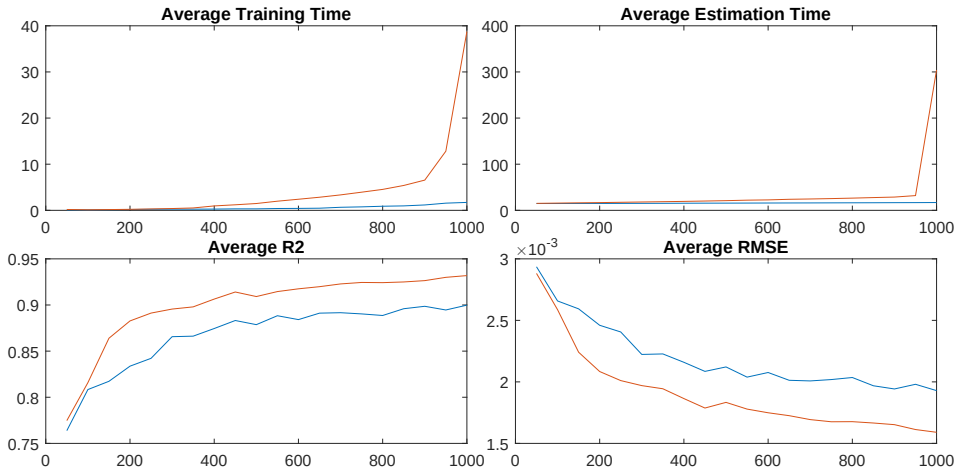


Figure 5.3: Global sensitivity measure estimation performance for LevelE code using fast Kriging and UQLab Kriging subroutines. Training size increases from 100 to 1000. The blue lines stand for the fast Kriging estimates, the red for UQlab. The average RMSE and R^2 are calculated based on $n_{pred} = 3,000$ predictions.

about 15 seconds. That is, with relative small training size $n_{train} = 400$, fast Kriging provides a saving of about 20%. Increasing training size to $n_{train} = 1000$, the saving time of fast Kriging builds up to 94%.

We now come to STOCFOR3. For this simulator, sensitivity measures have not been computed in previous studies due to the 'curse of dimensionality'; in fact, STOCFOR3 has slightly over forty thousand $(40,216)$ inputs. To the authors recollection, the code of largest dimensionality on which global sensitivity measures have been estimated is the probabilistic safety assessment code developed by the Idaho National Engineering Laboratory for the NASA constellation space mission project. Such simulator registers $d = 870$ inputs and has been used in Plischke et al. (2013). Thus, STOCFOR3 is a simulator of notably larger dimensionality. We employ the same strategy as before considering $n_{train} = [1000, 5000, 10000, 15000, 20000]$.

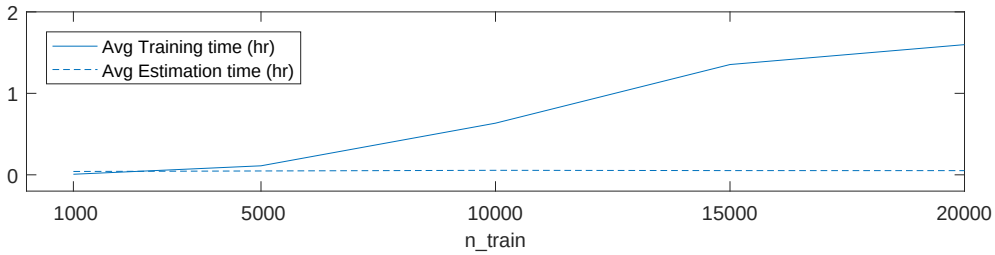


Figure 5.4: Global sensitivity measure estimation performance for STOCFOR3 using fast Kriging, $n_{train} = [1000, 5000, 10000, 15000, 20000]$. Blue line represents the average training time over 30 replicates; dotted line represents the average estimation time.

Results for the average training and estimation times are reported in Figure 5.4. The

training time increases from 25.18 secs at $n_{train} = 1000$ to 1.59 hours at $n_{train} = 20,000$; the time for estimating the global sensitivity measures remains about 0.05 hours (dotted line in Figure 5.4).

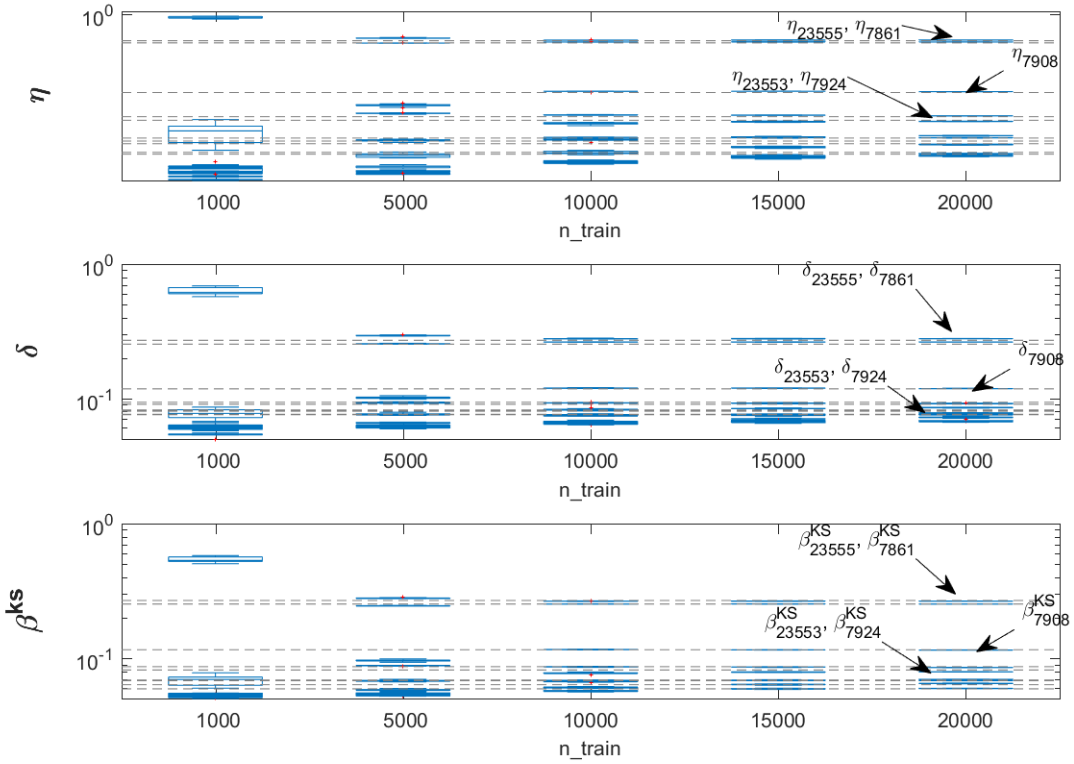


Figure 5.5: Global sensitivity measure estimates for STOCFOR3 using fast Kriging, $n_{train} = [1000, 5000, 10000, 15000, 20000]$, $m = 0.6n_{train}$.

Figure 5.5 displays the estimates of global sensitivity measures η_i, δ_i and β_i^{KS}, as n_{train} increases. Error bands are obtained with 30 replicates. One observes that, as n_{train} increases, the estimates of global sensitivity measures become stable and Error bands collapse.

Note that, there are (only) five main uncertainty drivers, namely, X^{23555}(RHS), X^{7861} (coefficient), X^{7908} (RHS), X^{23553} (coefficient) and X^{7924} (RHS). Inputs X^{23555} and X^{7861} have a similar importance. The estimated values of the variance-based sensitivity measures (η_i) show that they contribute to about 41% and 39% of the output variance respectively. The values of the density-based ($\delta_{23555} = 0.2779$ and $\delta_{7861} = 0.2621$) and distribution-based sensitivity measures ($\beta^{KS}_{23555} = 0.2652$ and $\beta^{KS}_{7861} = 0.2524$), confirm their relevance. The sensitivity measures estimates of these two inputs are notably higher than the sensitivity measures of the inputs ranking from 3$^{\text{rd}}$ to 5$^{\text{th}}$. For simulator input X^{7908} (3$^{\text{rd}}$), the sensitivity measure estimates are $\eta_{7908} = 0.0742, \delta_{7908} = 0.1178$ and $\beta^{KS}_{7908} = 0.1150$. For inputs X^{23553} and X^{7924} (4$^{\text{th}}$ and 5$^{\text{th}}$), the estimates are $\eta_{23553} = 0.0329, \delta_{23553} = 0.0913, \beta^{KS}_{23553} = 0.0854$ and $\eta_{7924} = 0.0274, \delta_{7924} = 0.0850, \beta^{KS}_{7924} = 0.0793$,

respectively. The remaining simulator inputs have sensitivity measures of lower values.

The above results suggest the following. Although dealing with a dimensionally large problem, the analyst can restrict her/his attention to only 5 inputs over $40,000$ that dominate uncertainty in the simulator response, at least in a first iteration. From a general perspective, we encounter a phenomenon similar to what is stated in Kleijnen (2008, Section 4.1), and Tolk et al. (2017, Section 8.2), and called the *Pareto* principle, suggesting that in business and economics applications only a few inputs may play a key role.

5.8 Summary

We have merged an innovation in machine learning with a traditional tool in simulations to obtain a *fast Kriging* method that reduces computational time and memory usage. The theoretical and algorithmic properties of the method have been analyzed. We have shown that the algorithm requires a total cost of $O(nm(d+m))$ in time, $O(m^2+n)$ in space.

In a series of numerical experiments of increasing dimensionality, we have compared fast Kriging with other Kriging subroutines in use. We have tested performance in training through traditional performance measures (e.g., RMSE); we have also challenged the prediction precision using the estimation of probabilistic sensitivity measures, which are complex functionals of the simulator output, and are well-known to be computationally challenging. The experiments show that fast Kriging allows the analyst to deal with larger samples, reduces computational time and preserves accuracy in fitting and prediction. In particular, fast Kriging permits the emulation of a dimensionally large simulators allowing one to breach the $10,000$ simulator input wall.

Chapter 6

Summary and future work

Computer experiments have been widely used in supporting scientific investigations. Sensitivity analysis becomes challenging when the computer code is complicated. The difficulties may come from the long evaluation-time or the high dimensionality of the simulator. This thesis has developed methods to cope with these issues, mainly focusing on global sensitivity measure estimation using Bayesian non-parametric techniques and Kriging meta-modeling that reduces the computational complexity in terms of time and memory requirements while achieving the same accuracy.

In chapter 2, we thoroughly investigated the relevant literature in sensitivity analysis. Sensitivity analysis techniques are categorized into the classes of local and global. For local methods, we present the one-at-a-time approach and differential-based sensitivity methods. For global methods, we illustrate the regression-based and variance-based sensitivity indices, followed by a common rationale of global sensitivity measures, following by the nullity-implies-independence and monotonic transformation invariant properties.

In chapter 3, we conduct a systematic sensitivity analysis on complex hydrological models. In particular, we focus on sensitivity methods that provide insights of the input-output mapping simultaneously from a given set of simulation realizations, so that the computational burden is under control.

In chapter 4, we focus on the task of quantifying uncertainty in the estimates of global sensitivity measures at small sample sizes. A Bayesian paradigm is proposed for the estimation, adopting advanced Bayesian non-parametric techniques. We have developed four classes of Bayesian estimators, among which, two classes bypass the partition selection issues by using Bayesian non-parametric joint and conditional density estimation. These two classes of estimators provide credibility intervals without requiring additional simulation runs. Possible future research directions regrading this work are listed here.

- The numerical implementation of the estimation of the sensitivity measures based on BNP density estimation and BNP density regression requires a combination of numerical integration and MCMC. The current algorithms may not be efficient

enough for large datasets, say $n > 1000$. Thus, MCMC posterior sampling with improved efficiency would be an exciting direction for future research.

- Bayesian non-parametric mixture models have advantages in flexibility and ability to capture the complex structures that are likely to be present in the dataset. In this work, for simplicity, Gaussian kernels have been considered. However, in practice, the analyst may deal with various types of distributions, the use of Gaussian kernels may not be appropriate in those cases. In this respect, a study of Bayesian non-parametric mixture models using other types of kernels (e.g. Beta or Log-Normal) is also an interesting avenue of future research.

- Because Bayesian non-parametric density estimation techniques provide the posterior distributions of the output response, either joint or conditional, there is potential to use these techniques as emulators to support a broader range of applications, for example, higher-order sensitivity measures estimations, simulation-based reliability assessment and sampling-based optimization.

In chapter 5, we focused on the development of a Kriging meta-model for dimensionally large simulators. Recent advances in machine learning are adopted. The proposed emulator reduces computational complexity regarding time and memory requirements while achieving the same accuracy. We have examined the performance of the proposed algorithm on several complex simulators, compared with commonly used emulators. We further applied the proposed emulator to estimate global sensitivity measures. In the future, research may focus on the following aspects.

- In recent studies, stochastic simulators have drawn increasing attention (Nelson, 2004; Xie et al., 2014). However, classical Kriging, as an interpolation method, is designed for deterministic simulators, and may not be appropriate for stochastic simulators. Recently, stochastic Kriging meta-modeling is becoming an active research area (Yin et al., 2009; Ankenman et al., 2010; Chen et al., 2012; Picheny et al., 2013). Ankenman et al. (2010) propose a stochastic Kriging meta-model which takes into account both uncertainty about the response surface of interest and the random simulation noise. Similar to classical Kriging, the stochastic Kriging also suffers the issue of computational complexity (the inversion of a $n \times n$ matrix). A study that reduces the computational complexity of the stochastic Kriging is to be carried out.

- Design of experiments is a widely developed discipline that aims to support the data collection stage to ensure the fidelity of simulators. Space-filling designs are suggested for complex simulators involving systematic randomness (Sacks et al., 1989). Koehler and Owen (1996) summarize several frequentist and Bayesian space filling designs, such as minimax and maximin designs (Johnson et al., 1990), maximum entropy designs (Currin et al., 1991), mean-squared-error designs, scrambled

nets, orthogonal arrays, and Latin Hypercube designs. An empirical comparison among those designs can be found in Simpson et al. (2001). The development of sampling designs that are specifically optimal for Kriging and stochastic Kriging is an interesting avenue of further research.

Bibliography

Alis, O. and H. Rabitz (2001). Efficient implementation of high-dimensional model representations. *Journal of Mathematical Chemistry 29*(2), 127–142.

Anderson, B., E. Borgonovo, M. Galeotti, and R. Roson (2014). Uncertainty in climate change modelling: Can global sensitivity analysis be of help? *Risk Analysis 34*(2), 271–293.

Ankenman, B., B. L. Nelson, and J. Staum (2010). Stochastic kriging for simulation metamodeling. *Operations Research Publication 58*(2), 371–382.

Antoniano-Villalobos, I., S. Wade, and S. G. Walker (2014). A Bayesian nonparametric regression model with normalized weights: A study of hippocampal atrophy in Alzheimer's disease. *Journal of the American Statistical Association 109*(506), 477–490.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American mathematical society 68*(3), 337–404.

Aster, R. C., B. Borchers, and C. H. Thurber (2013). *Parameter estimation and inverse problems*. Amsterdam, Academic Press.

Baker, C. T. (1977). The numerical treatment of integral equations.

Barrientos, A. F., A. Jara, and F. A. Quintana (2012). On the support of MacEachern's dependent Dirichlet processes and extensions. *Bayesian Analysis 7*, 277–310.

Barton, R. R. and M. Meckesheimer (2006). Metamodel-based optimization. *Handbooks in operations research and management science 13*, 535–574.

Baucells, M. and E. Borgonovo (2013). Invariant probabilistic sensitivity analysis. *Management Science 59*(11), 2536–2549.

Beccacece, F. and E. Borgonovo (2011). Functional ANOVA, ultramodularity and monotonicity: Applications in multiattribute utility theory. *European Journal of Operational Research 210*(2), 326–335.

Becker, W., S. Tarantola, and G. Deman (2018). Sensitivity analysis approaches to high-dimensional screening problems at low sample size. *Journal of Statistical Computation and Simulation 88*(11), 2089–2110.

Bettonvil, B. (1990). *Detection of important factors by sequential bifurcation*. Tilburg University Press, Tilburg.

Bettonvil, B. and J. P. Kleijnen (1997). Searching for important factors in simulation models with many factors: Sequential bifurcation. *European Journal of Operational Research 96*(1), 180–194.

Beven, K. J. (2011). *Rainfall-runoff modelling: The primer*. John Wiley & Sons.

Beven, K. J. and A. M. Binley (1992). The future of distributed models: Model calibration and predictive uncertainty. *Hydrological Processes*, 279–298.

Blackwell, D. and J. B. MacQueen (1973). Ferguson distributions via Pólya urn schemes. *The Annals of Statistics 1*(2), 353–355.

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

Borgonovo, E. (2007). A new uncertainty importance measure. *Reliability Engineering & System Safety 92*(6), 771–784.

Borgonovo, E. (2010). Sensitivity analysis with finite changes: An application to modified EOQ models. *European Journal of Operational Research 200*(1), 127–138.

Borgonovo, E. (2017). *Sensitivity Analysis: An Introduction for the Management Scientist* (Springer I ed.). Springer New York.

Borgonovo, E. and G. E. Apostolakis (2001). A new importance measure for risk-informed decision making. *Reliability Engineering & System Safety 72*(2), 193–212.

Borgonovo, E., W. Castaings, and S. Tarantola (2012). Model emulation and moment-independent sensitivity analysis: An application to environmental modeling. *Environmental Modelling and Software 34*, 105–115.

Borgonovo, E., G. Hazen, and E. Plischke (2016). A common rationale for global sensitivity measures and their estimation. *Risk Analysis 36*(10), 1871–1895.

Borgonovo, E., X. Lu, E. Plischke, O. Rakovec, and M. C. Hill (2017). Making the most out of a hydrological model data set: Sensitivity analyses to open the model black-box. *Water Resources Research 53*(9), 7933–7950.

Borgonovo, E. and E. Plischke (2016). Sensitivity analysis: A review of recent advances. *European Journal of Operational Research 3*(1), 869–887.

Borgonovo, E. and S. Tarantola (2008). Moment independent and variance-based sensitivity analysis with correlations. *International Journal of Chemical Kinetics 40*(11), 687–698.

Borgonovo, E., S. Tarantola, E. Plischke, and M. D. Morris (2014). Transformation and invariance in the sensitivity analysis of computer experiments. *Journal of the Royal Statistical Society Series B 76*(5), 925–947.

Butler, M. P., P. M. Reed, K. Fisher-Vanden, K. Keller, and T. Wagener (2014a). Identifying parametric controls and dependencies in integrated assessment models using global sensitivity analysis. *Environmental Modelling & Software 59*, 10–29.

Butler, M. P., P. M. Reed, K. Fisher-Vanden, K. Keller, and T. Wagener (2014b). Inaction and climate stabilization uncertainties lead to severe economic risks. *Climatic Change 127*(3-4), 463–474.

Camerlenghi, F., A. Lijoi, P. Orbanz, and I. Prünster (2018). Distribution theory for hierarchical processes.

Camerlenghi, F., A. Lijoi, and I. Prünster (2017). Bayesian prediction with multiple-samples information. *Journal of Multivariate Analysis 156*, 18–28.

Campolongo, F., J. P. Kleijnen, and T. Andres (2000). Screening methods. In A. Saltelli, K. Chan, and E. M. Scott (Eds.), *Sensitivity Analysis*, Chapter 4, pp. 65–80.

Campolongo, F. and A. Saltelli (1997). Sensitivity analysis of an environmental model: An application of different analysis methods. *Reliability Engineering & System Safety 57*(1), 49–69.

Caponnetto, A. and E. De Vito (2007). Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics 7*(3), 331–368.

Castaings, W., E. Borgonovo, S. Tarantola, and M. D. Morris (2012). Sampling strategies in density-based sensitivity analysis. *Environmental Modelling & Software 38*, 13–26.

Chatterjee, S. and A. S. Hadi (2006). *Regression Analysis by Example*, Volume 607.

Chen, V. C., K.-L. Tsui, R. R. Barton, and M. Meckesheimer (2006). A review on design, modeling and applications of computer experiments. *IIE Transactions 38*(4), 273–291.

Chen, X., B. E. Ankenman, and B. L. Nelson (2012). The effects of common random numbers on stochastic kriging metamodels. *ACM Transactions on Modeling and Computer Simulation 22*(2), 7.

Chen, X., B. E. Ankenman, and B. L. Nelson (2013). Enhancing stochastic Kriging metamodels with gradient estimators. *Operations Research 61*(2), 512–528.

Chen, X. and K. K. Kim (2016). Efficient VaR and CVaR measurement via stochastic kriging. *INFORMS Journal on Computing 28*(4), 629–644.

Clark, M. P., D. Kavetski, and F. Fenicia (2011). Pursuing the method of multiple working hypotheses for hydrological modeling. *Water Resources Research 47*(9), 1–16.

Clark, M. P., H. K. McMillan, D. B. G. Collins, D. Kavetski, and R. A. Woods (2011). Hydrological field data from a modeller's perspective: Part 2: Process-based evaluation of model hypotheses. *Hydrol. Processes 25*(4), 523–543.

Clark, M. P., B. Nijssen, J. D. Lundquist, D. Kavetski, D. E. Rupp, R. A. Woods, J. E. Freer, E. D. Gutmann, A. W. Wood, L. D. Brekke, and Others (2015). A unified approach for process-based hydrologic modeling: 1. Modeling concept. *Water Resources Research 51*(4), 2498–2514.

Clark, M. P., B. Nijssen, J. D. Lundquist, D. Kavetski, D. E. Rupp, R. A. Woods, J. E. Freer, E. D. Gutmann, A. W. Wood, D. J. Gochis, and Others (2015). A unified approach for process-based hydrologic modeling: 2. Model implementation and case studies. *Water Resources Research 51*(4), 2515–2542.

Clark, M. P., A. G. Slater, D. E. Rupp, R. A. Woods, J. A. Vrugt, H. V. Gupta, T. Wagener, and L. E. Hay (2008). Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models. *Water Resources Research 44*, 1–14.

Clarke, S. M., J. H. Griebsch, and T. W. Simpson (2005). Analysis of support vector regression for approximation of complex engineering analyses. *Journal of Mechanical Design 127*(6), 1077.

Cloke, H., F. Pappenberger, and J.-P. Renaud (2008). Multi-Method Global Sensitivity Analysis (MMGSA) for modelling floodplain hydrological processes. *Hydrological Processes 22*(11), 1660–1674.

Confalonieri, R., G. Bellocchi, S. Tarantola, M. Acutis, M. Donatelli, and G. Genovese (2010). Sensitivity analysis of the rice model WARM in Europe: Exploring the effects of different locations, climates and methods of analysis on model sensitivity to crop parameters. *Environmental Modelling and Software 25*(4), 479–488.

Conti, S., J. P. Gosling, J. E. Oakley, and A. O'Hagan (2009). Gaussian process emulation of dynamic computer codes. *Biometrika 96*(3), 663–676.

Critchfield, G. G. and K. E. Willard (1986). Probabilistic analysis of decision trees using Monte Carlo simulation. *Medical Decision Making 6*(2), 85–92.

Crnkovic, C. and J. Drachman (1996). Quality Control. *Risk 9*(9), 139–143.

Cuntz, M., J. Mai, M. Zink, S. Thober, R. Kumar, D. Schäfer, M. Schrön, J. Craven, O. Rakovec, D. Spieler, V. Prykhodko, G. Dalmasso, J. Musuuza, B. Langenberg, S. Attinger, and L. Samaniego (2015). Computationally inexpensive identification of noninformative model parameters by sequential screening. *Water Resources Research 51*(8), 6417–6441.

Currin, C., T. Mitchell, M. Morris, and D. Ylvisaker (1991). Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Source Journal of the American Statistical Association 86*(416), 953–963.

Dean, A. and S. Lewis (2006). *Screening: Methods for experimentation in industry, drug discovery, and genetics*.

Dellino, G., J. P. Kleijnen, and C. Meloni (2012). Robust optimization in simulation: Taguchi and Krige combined. *INFORMS Journal on Computing 24*(3), 471–484.

Dobler, C. and F. Pappenberger (2013). Global sensitivity analyses for a complex hydrological model applied in an alpine watershed. *Hydrological Processes 27*(26), 3922–3940.

Draper, N. R. and H. Smith (1998). *Applied regression analysis*. New York, Wiley.

Duan, Q., S. Sorooshian, and V. Gupta (1992). Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Resources Research 28*(4), 1015–1031.

Dunson, D. B. and J. H. Park (2008, jun). Kernel stick-breaking processes. *Biometrika 95*(2), 307–323.

Durrande, N., D. Ginsbourger, O. Roustant, and L. Carraro (2013). ANOVA kernels and RKHS of zero mean functions for model-based sensitivity analysis. *Journal of Multivariate Analysis 115*, 57–67.

Dyn, N., D. Levin, and S. Rippa (1986). Numerical procedures for surface fitting of scattered data by radial functions. *SIAM Journal on Scientific and Statistical Computing 7*(2), 639–659.

Efron, B. and G. Gong (1983). A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician 37*(1), 36–48.

Efron, B. and C. Stein (1981). The jackknife estimate of variance. *The Annals of Statistics 9*(3), 586–596.

Erickson, C. B., B. E. Ankenman, and S. M. Sanchez (2018). Comparison of Gaussian process modeling software. *European Journal of Operational Research 266*(1), 179–192.

Eschenbach, T. G. (1992). Spiderplots versus tornado diagrams for sensitivity analysis. *Interfaces 22*(6), 40–46.

Escobar, M. D. and M. West (1995). Bayesian density estimation and inference using mixtures. *Journal of the american statistical association 90*(430), 577–588.

European Commission (2009). Impact Assessment Guidelines.

Fang, H. and M. F. Horstemeyer (2006). Global response approximation with radial basis functions. *Engineering Optimization 38*(4), 407–424.

Favaro, S., A. Lijoi, and I. Prünster (2012). On the stick-breaking representation of normalized inverse Gaussian priors. *Biometrika 99*(3), 663–674.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics 1*(2), 209–230.

Ferguson, T. S. (1983). Bayesian density estimation by mixtures of normal distributions. In M. H. Rizvi, J. S. Rustagi, and D. Siegmund (Eds.), *Recent advances in statistics*, pp. 287–302. Academic Press.

Foglia, L., S. W. Mehl, M. C. Hill, and P. Burlando (2013). Evaluating model structure adequacy: The case of the Maggia Valley groundwater system, southern Switzerland. *Water Resources Research 49*(1), 260–282.

Forrester, A. I. J., A. Sóbester, and A. J. Keane (2008). *Engineering Design via Surrogate Modelling: A Practical Guide*. John Wiley & Sons, Ltd.

Freedman, D. and P. Diaconis (1981). On the histogram as a density estimator: L^2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete 57*(4), 453–476.

Freedman, D. A. (1965). On the asymptotic behavior of Bayes estimates in the discrete case II . *The Annals of Mathematical Statistics 36*(2), 454–456.

Friedman, J. H. (1991). Multivariate Adaptive Regression Splines. *The Annals of Statistics 19*(1), 1–67.

Fruth, J., O. Roustant, and S. Kuhnt (2014). Total interaction index: A variance-based sensitivity index for second-order interaction screening. *Journal of Statistical Planning and Inference 147*, 212–223.

Gamboa, F., A. Janon, T. Klein, A. Lagnoux, and C. Prieur (2016). Statistical inference for Sobol pick-freeze Monte Carlo method. *Statistics 50*(4), 881–902.

Gao, L., B. A. Bryan, M. Nolan, J. D. Connor, X. Song, and G. Zhao (2016). Robust global sensitivity analysis under deep uncertainty via scenario analysis. *Environmental Modelling & Software 76*, 154–166.

Ghanem, R., D. Higdon, and D. Owhadi (Eds.) (2016). *Handbook of Uncertainty Quantification*. Springer.

Gramacy, R. B. (2016). laGP: Large-scale spatial modeling via local approximate Gaussian processes in R. *Journal of Statistical Software 72*(1), 1–46.

Gramacy, R. B. and D. W. Apley (2013). Local Gaussian process approximation for large computer experiments.

Griewank, A. and A. Walther (2008). *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation* (Second ed.). SIAM.

Griffin, J. E. and M. F. J. Steel (2006). Order-based dependent Dirichlet processes. *Journal of the American Statistical Association 101*(473), 179–194.

Gupta, H. V., M. P. Clark, J. A. Vrugt, G. Abramowitz, and M. Ye (2012). Towards a comprehensive assessment of model structural adequacy. *Water Resources Research 48*(8), 1–16.

Hastie, T., R. Tibshirani, and J. Friedman (2009). The elements of statistical learning. *Elements 1*, 337–387.

Hastie, T. J. and R. J. Tibshirani (1990). *Generalized Additive Models*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis.

He, X. (2017). Rotated sphere packing designs. *Journal of the American Statistical Association 112*(520), 1612–1622.

Helton, J. C. and C. J. Sallaberry (2009). Computational implementation of sampling-based approaches to the calculation of expected dose in performance assessments for the proposed high-level radioactive waste repository at Yucca Mountain, Nevada. *Reliability Engineering & System Safety 94*(3), 699–721.

Hensman, J., N. Fusi, and N. D. Lawrence (2013). Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*.

Hill, M. C., L. Foglia, S. Christensen, O. Rakovec, and E. Borgonovo (2016). Model validation: Testing models using data and sensitivity analysis. In e. T. press). Cushman, J.H. and Tartakovsky, D.M. and F. I. . (in (Eds.), *Handbook of Groundwater Engineering*, Chapter 21. CRC Press, Taylor & Francis Group.

Hill, M. C., D. Kavetski, M. Clark, M. Ye, M. Arabi, D. Lu, L. Foglia, and S. Mehl (2016). Practical Use of Computationally Frugal Model Analysis Methods. *Groundwater 54*(2), 159–170.

Hill, M. C. and C. R. Tiedeman (2007). *Effective groundwater model calibration: with analysis of data, sensitivities, prediction and uncertainty*. Wiley.

Hjort, N. L. (1985). Bayesian nonparametric bootstrap confidence intervals. Technical report, DTIC.

Hjort, N. L. (1991). Bayesian and empirical Bayesian bootstrapping. Technical report, Matematisk Institutt, Universitetet i Oslo.

Hjort, N. L., C. Holmes, P. Müller, and S. G. Walker (Eds.) (2010). *Bayesian Nonparametrics*. Cambridge University Press.

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

Hong, L. J., B. L. Nelson, and J. Xu (2015). Discrete Optimization via Simulation. In *Handbook of Simulation Optimization*, Volume 1, pp. 9–44.

Hornberger, G. M. and R. C. Spear (1981). An approach to the preliminary analysis of environmental systems. *Journal of Environmental Management 7*, 7–18.

Houle, E. S., B. Livneh, and J. R. Kasprzyk (2017). Exploring snow model parameter sensitivity using Sobol' variance decomposition. *Environmental Modelling & Software 89*, 144–158.

Howard, R. A. (1988a). Decision analysis: practice and promise. *Management Science 34*(6), 679–695.

Howard, R. A. (1988b). Uncertainty about probability: A decision analysis perspective. *Risk Analysis 8*(1), 91–98.

Howarth, R. J. (1979). Mining Geostatistics. *Mineralogical Magazine 43*(328), 563–564.

Huang, D., T. T. Allen, W. I. Notz, and R. A. Miller (2006). Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization 32*(5), 369–382.

Iooss, B. and P. Lemaître (2015). A review on gobal snsitivity analysis methods. *Uncertainty management in Simulation-Optimization of Complex Systems: Algorithms and Applications*, 101–122.

Ishigami, T. and T. Homma (1990). An importance quantification technique in uncertainty analysis for computer models. In *Uncertainty modelling and analysis.*

Ishwaran, H. and L. F. James (2001). Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association 96*(453), 161–173.

Jalali, H., I. Van Nieuwenhuyse, and V. Picheny (2017). *Comparison of Kriging-based algorithms for simulation optimization with heterogeneous noise*, Volume 261. Elsevier B.V.

Janon, A., T. Klein, A. Lagnoux, M. Nodet, and C. Prieur (2014). Asymptotic normality and efficiency of two Sobol index estimators. *ESAIM:Probability and Statistics 18*, 342–364.

Janon, A., M. Nodet, and C. Prieur (2014). Uncertainties assessment in global sensitivity indices estimation from metamodels. *International Journal for Uncertainty Quantification 4*(1), 21–36.

Jara, A., T. E. Hanson, F. A. Quintana, P. Müller, and G. L. Rosner (2011). Dppackage: Bayesian semi-and nonparametric modeling in r. *Journal of Statistical Software 40*(5), 1.

Johnson, M. E., L. M. Moore, and D. Ylvisaker (1990). Minimax and maximin distance designs. *Journal of Statistical Planning and Inference 26*(2), 131–148.

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

Jungermann, H. and M. Thuring (1988). The labyrinth of experts' minds: Some reasoning strategies and their pitfalls. *Annals of Operations Research 16*, 117–130.

Kalli, M., J. E. Griffin, and S. G. Walker (2011). Slice sampling mixture models. *Statistics and Computing 21*(1), 93–105.

Khorashadi Zadeh, F., J. Nossent, F. Sarrazin, F. Pianosi, A. van Griensven, T. Wagener, and W. Bauwens (2017). Comparison of variance-based and moment-independent global sensitivity analysis approaches by application to the SWAT model. *Environmental Modelling and Software 91*, 210–222.

Kimeldorf, G. and G. Wahba (1971). Some results on Tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications 33*(1), 82–95.

Kleijnen, J. (2008). *Design and Analysis of Simulation Experiments*. Berlin: Springer Verlag.

Kleijnen, J. P. (2005). An overview of the design and analysis of simulation experiments for sensitivity analysis. *European Journal of Operational Research 164*(2), 287–300.

Kleijnen, J. P. (2009). Kriging metamodeling in simulation: A review. *European Journal of Operational Research 192*(3), 707–716.

Kleijnen, J. P. (2017). Regression and Kriging metamodels with their experimental designs in simulation: A review. *European Journal of Operational Research 256*(1), 1–16.

Kleijnen, J. P. and W. C. Van Beers (2005). Robustness of Kriging when interpolating in random simulation with heterogeneous variances: Some experiments. *European Journal of Operational Research 165*(3), 826–834.

Kleijnen, J. P. C. (2014). Simulation-optimization via Kriging and bootstrapping: a survey. *Journal of Simulation 8*(4), 241–250.

Koehler, J. R. and A. B. Owen (1996). 9 Computer experiments. *Handbook of statistics 13*, 261–308.

König, H. (1986). *Eigenvalue distribution of compact operators*. Birkhäuser.

Krige, D. G. (1952). A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand. *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, 201–215.

Kucherenko, S., M. Rodriguez-Fernandez, C. Pantelides, and N. Shah (2009). Monte carlo evaluation of derivative-based global sensitivity measures. *Reliability Engineering & System Safety 94*(7), 1135–1148.

Kuiper, N. H. (1960). Tests concerning random points on a circle. *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen, Series A 63*, 38–47.

Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li (1996). *Applied Linear Statistical Models*, Volume Fifth.

La Vigna, F., M. C. Hill, R. Rossetto, and M. R. (2016). Parameterization, sensitivity analysis, and inversion: an investigation using groundwater modeling of the surface-mined Tivoli-Guidonia basin (Metropolitan City of Rome, Italy). *Hydrogeology Journal forthcomin*, 1–19.

Lampinen, J. and a. Vehtari (2001). Bayesian approach for neural networks–review and case studies. *Neural networks : the official journal of the International Neural Network Society 14*(3), 257–274.

Lataniotis, C., S. Marelli, and B. Sudret (2015). UQLab user manual-Kriging (Gaussian process modelling). Technical report, Chair of Risk, Safety & Uncertainty Quantification, ETH Zurich.

Lawrence, N. D., M. Seeger, and R. Herbrich (2003). Fast sparse Gaussian process methods: The informative vector machine. *Advances in Neural Information Processing Systems 15*, 625–632.

LeCun, Y. A., Y. Bengio, and G. E. Hinton (2015). Deep learning. *Nature 521*(7553), 436–444.

Li, G. and H. Rabitz (2012). General Formulation of HDMR Component Functions with Independent and Correlated Variables. *Journal of Mathematical Chemistry 50*, 99–130.

Lijoi, A., R. H. Mena, and I. Prünster (2007). Controlling the reinforcement in Bayesian non-parametric mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 69*(4), 715–740.

Lin, C. D., D. Bingham, R. R. Sitter, and B. Tang (2010). A new and flexible method for constructing designs for computer experiments. *The Annals of Statistics 38*(3), 1460–1477.

Lo, A. Y. and Others (1984). On a class of Bayesian nonparametric estimates: I. Density estimates. *The Annals of Statistics 12*(1), 351–357.

Lophaven, S. N., J. Søndergaard, and H. B. Nielsen (2002). DACE: a Matlab kriging toolbox. *IMM Informatiocs and Mathematical Modelling*, 1–28.

Lu, D., M. Ye, and M. C. Hill (2012). Analysis of regression confidence intervals and Bayesian credible intervals for uncertainty quantification. *Water Resources Research 48*, 1–20.

Luo, J., L. J. Hong, B. L. Nelson, and Y. Wu (2015). Fully sequential procedures for large-scale ranking-and-selection problems in parallel computing environments. *Operations Research 63*(5), 1177–1194.

MacEachern, S. N. (1999). Dependent nonparametric processes. In *ASA proceedings of the section on Bayesian statistical science*, pp. 50–55. American Statistical Association.

MacEachern, S. N. (2000). Dependent Dirichlet processes. Technical report, Department of Statistics, The Ohio State University.

Marangoni, G., M. Tavoni, V. Bosetti, E. Borgonovo, P. Capros, O. Fricko, D. Gernaat, C. Guivarch, P. Havlik, D. Huppmann, N. Johnson, P. Karkatsoulis, I. Keppo, V. Krey, E. Ó Broin, J. Price, and D. P. Van Vuuren (2017). Sensitivity of projected long-term CO_2 emissions across the Shared Socioeconomic Pathways. *Nature Climate Change 7*(2), 113–117.

Marelli, S. and B. Sudret (2015). UQLab user manual-Polynomial Chaos Expansions. Technical report, Chair of Risk, Safety & Uncertainty Quantification, ETH Zurich (http://www.uqlab.com/).

Markstrom, S. L., L. E. Hay, and M. P. Clark (2016). Towards simplification of hydrologic modeling: identification of dominant processes. *Hydrology and Earth System Sciences 20*, 4655–4671.

Marrel, A., B. Iooss, S. Veiga, and M. Ribatet (2012). Global sensitivity analysis of stochastic computer models with joint metamodels. *Statistics and Computing 22*(3), 833–847.

Matheron, G. (1975). *Random sets and integral geometry*, Volume Wiley. Wiley New York.

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

Mendoza, P. A., M. P. Clark, M. Barlage, B. Rajagopalan, L. Samaniego, G. Abramowitz, and H. Gupta (2015). Are we unnecessarily constraining the agility of complex process-based models? *Water Resources Research 51*(1), 716–728.

Mitchell, T. J. and M. D. Morris (1992). The spatial correlation function approach to response surface estimation. *Proceedings of the 24th conference on Winter simulation Conference*, 565–571.

Mizukami, N., M. Clark, A. Newman, A. Wood, E. Gutmann, B. Nijssen, O. Rakovec, and L. Samaniego (2017). Towards seamless large domain parameter estimation for hydrologic models. *Water Resources Research*.

Montanari, A. (2007). What do we mean by uncertainty? The need for a consistent wording about uncertainty assessment in hydrology. *Hydrol. Processes 21*(6), 841–845.

Morris, M. (1993). Telling tails explain the discrepancy sexual partner reports. *Letters to nature, Nature 365*, 437.

Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics 33*(2), 161–174.

Müller, P. and F. A. Quintana (2004). Nonparametric Bayesian data analysis. *Statistical Science 19*(1), 95–110.

Nearing, G. S., Y. Tian, H. V. Gupta, M. P. Clark, K. W. Harrison, and S. V. Weijs (2016). A philosophical basis for hydrological uncertainty. *Hydrological Sciences Journal 61*(9), 1666–1678.

Neidinger (2010). Introduction to automatic differentiation and MATLAB object-oriented programming. *SIAM Review 52*(3), 545–563.

Nelson, B. (2004). Stochastic simulation research in management science. *Management Science 50*(7), 855–868.

Nesterov, Y. (2012). Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization 22*(2), 341–362.

NICE (2013). Guide to the methods of technology appraisal 2013.

Norton, J. (2015). An introduction to sensitivity assessment of simulation models. *Environ. Mod. & Soft. 69*, 166–174.

Nuclear Energy Agency (1989). Psacoin level E intercomparison. Technical report, OECD, Paris.

O' Brien, F. A. (2004). Scenario planning - lessons for practice from teaching and learning. *European Journal of Operational Research 152*, 709–722.

Oakley, J. and A. O'Hagan (2002). Bayesian inference for the uncertainty distribution of computer model outputs. *Biometrika 89*(4), 769–784.

Oakley, J. and A. O'Hagan (2004). Probabilistic sensitivity analysis of complex models: A Bayesian approach. *Journal of the Royal Statistical Society, Series B 66*(3), 751–769.

Paleari, L. and R. Confalonieri (2016). Sensitivity analysis of a sensitivity analysis: We are likely overlooking the impact of distributional assumptions. *Ecological Modelling 340*, 57–63.

Papaspiliopoulos, O. and G. O. Roberts (2008). Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models. *Biometrika 95*(1), 169–186.

Pappenberger, F. and K. J. Beven (2006). Ignorance is bliss: Or seven reasons not to use uncertainty analysis. *Water Resources Research 42*(5).

Pappenberger, F., K. J. Beven, M. Ratto, and P. Matgen (2008). Multi-method global sensitivity analysis of flood inundation models. *Advances in Water Resources 31*(1), 1–14.

Pearson, K. (1905). *On the General Theory of Skew Correlation and Non-linear Regression*, Volume XIV of *Mathematical Contributions to the Theory of Evolution, Drapers' Company Research Memoirs*. London: Dulau & Co.

Peckham, S. D., A. Kelbert, M. C. Hill, and E. W. H. Hutton (2016). Towards uncertainty quantification and parameter estimation for Earth system models in a component-based modeling framework. *Computers & Geosciences 90*, 152–161.

Pianosi, F., K. Beven, J. Freer, J. W. Hall, J. Rougier, D. B. Stephenson, and T. Wagener (2016). Sensitivity analysis of environmental models: A systematic review with practical workflow. *Environmental Modelling and Software 79*, 214–232.

Pianosi, F. and T. Wagener (2015). A simple and efficient method for global sensitivity analysis based oncumulative distribution functions. *Environmental Modelling and Software 67*, 1–11.

Picheny, V., T. Wagner, and D. Ginsbourger (2013). A benchmark of kriging-based infill criteria for noisy optimization. *Structural and Multidisciplinary Optimization 48*(3), 607–626.

Pitman, J. (1996). Random Discrete Distributions Invariant under Size-Biased Permutation. *Advances in Applied Probability 28*(2), 525–539.

Pitman, J. and M. Yor (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *The Annals of Probability*, 855–900.

Plischke, E. (2010). An effective algorithm for computing global sensitivity indices (EASI). *Reliability Engineering & System Safety 95 (4)*, 354–360.

Plischke, E. (2012). An Adaptive Correlation Ratio Method Using the Cumulative Sum of the Reordered Output. *Reliability Engineering & System Safety 107*, 149–156.

Plischke, E. and E. Borgonovo (2017). Probabilistic Sensitivity Measures from Empirical Cumulative Distribution Functions. *Work in Progress*.

Plischke, E., E. Borgonovo, and C. L. Smith (2013). Global Sensitivity Measures from Given Data. *European Journal of Operational Research 226*(3), 536–550.

Preuss, M., T. Wagner, and D. Ginsbourger (2012). High-dimensional model-based optimization based on noisy evaluations of computer games. *Lecture Notes in Computer*

Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 7219 LNCS, 145–159.

Qu, H. and M. C. Fu (2014). Gradient Extrapolated Stochastic Kriging. *ACM Transactions on Modeling and Computer Simulation 24*(4), 1–25.

Quiñonero-candela, J., C. E. Rasmussen, and R. Herbrich (2005). A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research 6*, 1935–1959.

Rabitz, H. and O. F. Alis (1999). General foundations of High-Dimensional Model Representations. *J. Math. Chem. 25*(2-3), 197–233.

Rahman, S. (2014). A Generalized ANOVA Dimensional Decomposition for Dependent Probability Measures. *SIAM/ASA Journal on Uncertainty Quantification 2*(1), 670–697.

Rahman, S. (2016). The f-Sensitivity Index. *SIAM/ASA Journal on Uncertainty Quantification 4*(1), 130–162.

Rakovec, O., M. C. Hill, M. P. Clark, A. H. Weerts, A. J. Teuling, and R. Uijlenhoet (2014). Distributed evaluation of local sensitivity analysis (DELSA), with application to hydrologic models. *Water Resources Research 50*(1), 409–426.

Rakovec, O., A. H. Weerts, P. Hazenberg, P. J. J. F. Torfs, and R. Uijlenhoet (2012). State updating of a distributed hydrological model with Ensemble Kalman Filtering: effects of updating frequency and observation network density on forecast accuracy. *Hydrol. Earth Syst. Sci. 16*, 3435–3449.

Ratto, M. and A. Pagano (2010). Using Recursive Algorithms for the Efficient Identification of Smoothing Spline ANOVA Models. *Advances in Statistical Analysis 94*, 367–388.

Ratto, M., A. Pagano, and P. Young (2007). State Dependent Parameter metamodelling and sensitivity analysis. *Computer Physics Communications 177*(11), 863–876.

Ratto, M., P. C. Young, R. Romanowicz, F. Pappenberger, A. Saltelli, and A. Pagano (2007). Uncertainty, sensitivity analysis and the role of data based mechanistic modeling in hydrology. *Hydrology and Earth System Sciences 11*(4), 1249–1266.

Razavi, S. and H. V. Gupta (2015). What do we mean by sensitivity analysis? The need for comprehensive characterization of "global" sensitivity in Earth and Environmental systems models. *Water Resources Research 51*(5), 3070–3092.

Razavi, S. and H. V. Gupta (2016a). A new framework for comprehensive, robust, and efficient global sensitivity analysis: 1. Theory. *Water Resources Research 52*(1), 423–439.

Razavi, S. and H. V. Gupta (2016b). A new framework for comprehensive, robust, and efficient global sensitivity analysis: 1. theory. *Water Resources Research 52*(1), 423–439.

Razavi, S., B. A. Tolson, and D. H. Burn (2012). Review of Surrogate Modeling in Water Resources. *Water Resources Research 48*(7), 1–32.

Renyi, A. (1959). On Measures of Statistical Dependence. *Acta Mathematica Academiae Scientiarum Hungarica 10*, 441–451.

Ripley, B. (1994). Neural networks and related methods for classification. *Journal of the Royal Statistical Society. Series B (Methodological) 56*, 409–456.

Rodriguez, A. and D. B. Dunson (2011). Nonparametric Bayesian models through probit stick-breaking processes. *Bayesian Analysis 6*, 145–178.

Roehlig, K., E. Plischke, R. Bolado Lavin, D. Becker, P. Ekstroem, and S. Hotzel (2009). Lessons learnt from studies on sensitivity analysis techniques in the EU project PAMINA: a benchmark study. In *Reliability, Risk and Safety. Theory and Applications. Proceedings of the ESREL 2009 Annual Conference*, Volume 3, Boca Raton, pp. 1769–1775. CRC Press.

Rojas, R. (1996). Neural networks: a systematic introduction. *Neural Networks*, 502.

Rosero, E., Z.-L. Yang, T. Wagener, L. E. Gulden, S. Yatheendradas, and G.-Y. Niu (2010). Quantifying parameter sensitivity, interaction, and transferability in hydrologically enhanced versions of the noah land surface model over transition zones during the warm season. *Journal of Geophysical Research Atmospheres 115*(3).

Rudi, A., R. Camoriano, and L. Rosasco (2015). Less is More : Nyström Computational Regularization. *Advances in Neural Information Processing Systems 28 (NIPS 2015)*, 1–9.

Sacks, J., R. S. Schiller, and W. J. Welch (1989). Designs for computer experiments. *Technometrics 31*, 41–47.

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

Saltelli, A. (2002a). Making Best Use of Model Valuations to Compute Sensitivity Indices. *Computer Physics Communications 145*, 280–297.

Saltelli, A. (2002b). Sensitivity Analysis for Importance Assessment. *Risk Analysis 22*(3), 579–590.

Saltelli, A. and P. Annoni (2010). How to avoid a perfunctory sensitivity analysis. *Environmental Modeling and Software 25*, 1508–1517.

Saltelli, A., K. Chan, and E. M. Scott (2000). *Sensitivity Analysis*. Chichester.

Saltelli, A. and B. D'Hombres (2010). Sensitivity analysis didn't help. A practitioner's critique of the Stern review. *Global Environmental Change 20 (2)*, 298–302.

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

Saltelli, A., M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola (2008). *Global Sensitivity Analysis. The Primer*. John Wiley & Sons, Ltd.

Saltelli, A., M. Ratto, S. Tarantola, and F. Campolongo (2006). Sensitivity analysis practices: Strategies for model-based inference. *Reliability Engineering & System Safety 91*(10-11), 1109–1125.

Saltelli, A., M. Ratto, S. Tarantola, and F. Campolongo (2012). Update 1 of: Sensitivity Analysis for Chemical Models. *Chemical Reviews 112*(5), 1–21.

Saltelli, A. and S. Tarantola (2002). On the Relative Importance of Input Factors in Mathematical Models: Safety Assessment for Nuclear Waste Disposal. *Journal of the American Statistical Association 97*(459), 702–709.

Saltelli, A., S. Tarantola, and F. Campolongo (2000). Sensitivity Analysis as an Ingredient of Modelling. *Statistical Science 19*(4), 377–395.

Saltelli, A., S. Tarantola, F. Campolongo, and M. Ratto (2004). *Sensitivity Analysis in Practise – A Guide to Assessing Scientific Models*. Chichester.

Saltelli, A., S. Tarantola, and K. P. S. Chan (1999). A Quantitative Model-Independent Method for Global Sensitivity Analysis of Model Output. *Technometrics 41*(1), 39–56.

Samaniego, L., R. Kumar, S. Thober, O. Rakovec, M. Zink, N. Wanders, S. Eisner, H. Müller Schmied, E. H. Sutanudjaja, K. Warrach-Sagi, and S. Attinger (2017).

Toward seamless hydrologic predictions across scales. *Hydrol. Earth Syst. Sci. Discuss. 2017*, 1–36.

Samuelson, P. A. (1941). The Stability of Equilibrium: Comparative Statics and Dynamics. *Econometrica 9*(2), 97–120.

Santner, T., B. Williams, and W. Notz (2003). *The Design and Analysis of Computer Experiments*.

Schobi, R., B. Sudret, and J. Wiart (2015). Polynomial-Chaos-Based Kriging. *International Journal for Uncertainty Quantification 5*(2), 171–193.

Schölkopf, B., B. Scholkopf, and A. Smola (2002). Learning with kernels. *Journal of the Electrochemical Society 129*(November), 2865.

Seber, G. A. F. and C. J. Wild (1989). *Nonlinear regression*. Wiley, New York.

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica sinica 4*(2), 639–650.

Silvestrini, R. T., D. C. Montgomery, and B. Jones (2013). Comparing computer experiments for the Gaussian process model using integrated prediction variance. *Quality Engineering 25*(2), 164–174.

Simpson, T. W., J. D. Peplinski, P. N. Koch, and J. K. Allen (2001). Metamodels for computer-based engineering design: Survey and recommendations.

Smith, R. L., C. Tebaldi, D. Nychka, and L. O. Mearns (2009). Bayesian Modeling of Uncertainty in Ensembles of Climate Models. *Journal of the American Statistical Association 104*(485), 97–116.

Snelson, E. and Z. Ghahramani (2006). Sparse Gaussian Processes using Pseudo-inputs. *Advances in Neural Information Processing Systems 18*, 1257–1264.

Sobol', I. M. (1993). Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiment 1*, 407–414.

Sobol, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and computers in simulation 55*(1), 271–280.

Sobol', I. M. and S. Kucherenko (2009). Derivative Based Global Sensitivity Measures and their Links with Global Sensitivity Indices. *Mathematics and Computers in Simulation 79*, 3009–3017.

Storlie, C. B., L. P. Swiler, J. C. Helton, and C. J. Sallaberry (2009). Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models. *Reliability Engineering & System Safety 94*(11), 1735–1763.

Strong, M. and J. E. Oakley (2013). An efficient method for computing partial expected value of perfect information for correlated inputs. *Medical Decision-Making 33*(6), 755–766.

Strong, M., J. E. Oakley, and A. Brennan (2014). Estimating Multiparameter Partial Expected Value of Perfect Information from a Probabilistic Sensitivity Analysis Sample: A Nonparametric Regression Approach. *Medical Decision-Making 34*, 311–326.

Strong, M., J. E. Oakley, A. Brennan, and P. Breeze (2015). Estimating the Expected Value of Sample Information Using the Probabilistic Sensitivity Analysis Sample: A Fast, Nonparametric Regression-Based Method. *Medical Decision Making 35*(5), 570–583.

Strong, M., J. E. Oakley, and J. Chilcott (2012). Managing structural uncertainty in health economic decision models: a discrepancy approach. *Journal of the Royal Statistical Society: Series C (Applied Statistics) 61*(1), 25–45.

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansion. *93*, 964–979.

Sun, L., L. J. Hong, and Z. Hu (2014). Balancing Exploitation and Exploration in Discrete Optimization via Simulation Through a Gaussian Process-Based Search. *Operations Research 62*(6), 1416–1438.

Tang, Y., P. Reed, K. Van Werkhoven, and T. Wagener (2007). Advancing the identification and evaluation of distributed rainfall-runoff models using global sensitivity analysis. *Water Resources Research 43*(6).

Tang, Y., P. Reed, T. Wagener, and K. Van Werkhoven (2007). Comparing sensitivity analysis methods to advance lumped watershed model identification and evaluation. *Hydrology and Earth System Sciences 11*(2), 793–817.

Teh, Y. W. and M. I. Jordan (2010). Hierarchical Bayesian Nonparametric Models with Applications. In N. Hjort, C. Holmes, P. Müller, and S. Walker (Eds.), *Bayesian Nonparametrics: Principles and Practice*. Cambridge University Press.

Teh, Y. W., M. I. Jordan, M. J. Beal, and D. M. Blei (2006). Hierarchical Dirichlet Processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

Tietje, O. (2005). Identification of a Small Reliable and Efficient Set of Consistent Scenarios. *European Journal of Operational Research 162*(4), 418–432.

Tissot, J.-Y. and C. Prieur (2015). A randomized Orthogonal Array-based procedure for the estimation of first- and second-order Sobol' indices. *Journal of Statistical Computation and Simulation 85*(7), 1358–1381.

Tolk, A., J. Fowler, G. Shao, and E. Yucesan (2017). *Advances in Modeling and Simulation: Seminal Research from 50 Years of Winter Simulation Conferences*. Springer.

Tuo, R. and J. Wu (2015). Efficient Calibration of Imperfect Computer Codes. *Annals of Statistics 43*(6), 2331–2352.

UK Government Office for Science (2018). *Computational Modelling: Technological Features*.

Urtasun, R. and T. Darrell (2008). Sparse probabilistic regression for activity-independent human pose inference. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8. IEEE.

US EPA (2009). Guidance on the Development, Evaluation, and Application of Environmental Models.

Van Beers, W. C. and J. P. Kleijnen (2003). Kriging for interpolation in random simulation. *Journal of the Operational Research Society 54*(3), 255–262.

van Werkhoven, K., T. Wagener, P. Reed, and Y. Tang (2009). Sensitivity-guided reduction of parametric dimensionality for multi-objective calibration of watershed models. *Advances in Water Resources 32*(8), 1154–1169.

Ver Hoef, J. M. and N. Cressie (1993). Multivariable spatial prediction. *Mathematical Geology 25*(2), 219–240.

Wan, H., B. E. Ankenman, and B. L. Nelson. Improving the efficiency and efficacy of controlled sequential bifurcation for simulation factor screening. *INFORMS Journal on Computing 22*(3), 482–492.

Wang, L.-F. and L.-Y. Shi (2013). Simulation optimization: a review on theory and applications. *Zidonghua Xuebao/Acta Automatica Sinica 39*(11), 1957–1968.

Wang, R.-T. (2007). A reliability model for multivariate exponential distributions. *Journal of Multivariate Analysis 98*(5), 1033–1042.

Wei, P., Z. Lu, and J. Song (2014). Moment-independent sensitivity analysis using copula. *Risk Analysis 34*(2), 210–222.

Welch, W. J., R. J. Buck, J. Sacks, H. P. Wynn, J. Toby, and M. D. Morris (1992). Screening, predicting, and computer experiments. *Technometrics 34*(1), 15–25.

Wendell, R. E. (2004). Tolerance sensitivity and optimality bounds in linear programming. *Management Science 50*(6), 797–803.

Wendland, H. (2004). *Scattered data approximation*. Cambridge university press.

Williams, C. and M. Seeger (2001). Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems 13*, pp. 682–688.

Williams, C. K. and C. E. Rasmussen (2006). *Gaussian Processes for Machine Learning*. MIT press Cambridge.

Wong, R. K. W., C. B. Storlie, and T. C. M. Lee (2017). A frequentist approach to computer model calibration. *Journal of the Royal Statistical Society. Series B: Statistical Methodology 79*(2), 635–648.

Wood, F., J. Gasthaus, C. Archambeau, L. James, and Y. W. Teh (2011). The sequence memoizer. *Communications of the Association for Computing Machines 54*(2), 91–98.

Woods, D. C. and S. M. Lewis (2017). Design of experiments for screening. In *Handbook of Uncertainty Quantification*, pp. 1143–1185.

Xie, W., B. L. Nelson, and R. R. Barton (2014). A Bayesian framework for quantifying uncertainty in stochastic simulation. *Operations Research 62*(6), 1439–1452.

Xu, J. (2012). Efficient discrete optimization via simulation using stochastic kriging. In *Proceedings of the 2012 Winter Simulation Conference*, Number 2010, pp. 1–12. Winter Simulation Conference.

Yau, C., O. Papaspiliopoulos, G. O. Roberts, and C. Holmes (2011). Bayesian non-parametric hidden Markov models with applications in genomics. *Journal of the Royal Statistical Society. Series B (Statistical Methodology) 73*(1), 37–57.

Yin, J., S. H. Ng, and K. M. Ng (2009). A study on the effects of parameter estimation on Kriging model's prediction error in stochastic simulations. In *proceedings of the 2009 Winter Simulation Conference*, Number 1993, pp. 674–685.

Yin, J., S. H. Ng, and K. M. Ng (2011). Kriging metamodel with modified nugget-effect: The heteroscedastic variance case. *Computers and Industrial Engineering 61*(3), 760–777.

Young, P. C. (1999). Data-based mechanistic modelling, generalised sensitivity and dominant mode analysis. *Computer Physics Communication 117*, 113–129.

Ziehn, T. and A. S. Tomlin (2009). GUI-HDMR - A software tool for global sensitivity analysis of complex models. *Environmenal Modelling & Software 24*, 775–785.