Efficient global sensitivity analysis for high-dimensional outputs combining data-driven probability models and dimensionality reduction

WoongHee Jung and Alexandros A. Taflanidis*

University of Notre Dame, Department of Civil and Environmental Engineering and Earth Sciences

*Corresponding author: a.taflanidis@nd.edu

Abstract

This paper examines the efficient variance-based global sensitivity analysis (GSA), quantified by estimating first-/higher-order and total-effect Sobol' indices, for applications involving complex numerical models and high-dimensional outputs. Two different, recently developed, techniques are combined to address the associated challenges. Principal component analysis (PCA) is first considered as a dimensionality reduction technique. The GSA for the original output vector is then formulated by calculating variance and covariance statistics for the low-dimensional latent output space identified by PCA. These statistics are efficiently approximated by extending recent work on data-driven, probability model-based GSA (PM-GSA). The extension, constituting the main novel contribution of this work, pertains to the estimation of covariance statistics beyond the variance statistics examined in the original PM-GSA formulation. Specifically, a Gaussian mixture model (GMM) is developed to approximate the joint probability density function between some subset of the input vector, and each latent output, or each pair of latent outputs. The GMM is then utilized to estimate the aforementioned statistics. Results across two natural hazards engineering examples show that the dimensionality reduction and transformation of output space established through PCA do not impact the overall accuracy of the PM-GSA, and that the proposed implementation accommodates highly-efficient GSA estimates.

Keywords: sensitivity analysis; high-dimensional output; principal component analysis; Gaussian mixture; Sobol' indices.

1. Introduction

Global sensitivity analysis (GSA) plays a prominent role in engineering analysis and design [1-5]. Its objective is to quantify the importance of the different model inputs with respect to their impact on the model output [6], the latter representing the quantity of interest (QoI) in the problem formulation. This quantification provides valuable insights into the system behavior, and it can be used to guide various tasks, including dimensionality or uncertainty reduction and optimal decision making. Different approaches exist

Author personal Copy: https://doi.org/10.1016/j.ress.2022.108805

to perform GSA [6-11], and perhaps the most popular one is variance-based decomposition utilizing Sobol' indices [12]. This paper examines the efficient estimation of these indices for engineering applications that involve complex numerical models and high-dimensional outputs. Objective is the estimation of the sensitivity indices for each of the output components. Efficiency in this setting refers to both the estimation of the indices using only a small number of model simulations, to accommodate the larger computational burden these simulations entail, as well as to the need to perform the estimation for each of the QoIs, which might exceed a few thousands of outputs in certain applications. Furthermore, our interest is in approaches that do not require a specialized numerical simulation to support the GSA estimation rather can be performed using generic simulation data. Such approaches can be seamlessly integrated within existing computational workflows, offering GSA as supplementary information to the primary tasks accomplished through these workflows.

For individual QoIs, Sobol' indices can be computed using a double-loop Monte Carlo integration (MCI), separately performed for each examined index. Since the double-loop MCI involves a substantial computational effort, alternative formulations have been considered over the past two decades [13]. These formulations include: highly efficient sampling schemes to perform the MCI [14-16] establishing a total MCI burden that increases linearly with the dimension of the model input; estimation using design of experiments concepts [17]; approximations using samples from auxiliary density functions [18, 19]; approaches that replace the original model with a fast-to-compute surrogate model [20-24], even accommodating an entirely analytical estimation of the indices when the surrogate model corresponds to Polynomial Chaos Expansion [20]. The domain of applicability of these approaches depends on the flexibility provided for performing the global sensitivity analysis, with some of them (for example some MCI approaches) requiring a specialized set of model simulations, selected with an explicit objective to support the sensitivity analysis, which somewhat limits their applicability. Purely data-driven approaches that rely on the availability of a generic sample input/output set without placing any restrictions on the characteristics of this set can overcome this limitation. Towards this objective, Hu and Mahadevan [25] recently established a powerful generalized probability-model GSA (PM-GSA) framework to compute different types of Sobol' indices. The framework relies on the development of a probability model to approximate the joint probability density between each input (or subsets of inputs) and the examined QoI. The probability models examined in [25] included Gaussian mixture models, Gaussian copula models and a new Gaussian mixture copula model, and all of them were shown to provide good estimation accuracy, with some marginal preference offered [25] for the Gaussian mixture model based on its demonstrated robustness. Implementations in [25], and in most cases referenced in this paragraph that do not fall in the MCI category, considered only a single QoI, meaning that for multi-output cases they will need to be independently performed for each output. Should be also noted that each of these alternative approaches has its own advantages, and depending on the implementation setting, for example the dimensionality of the input or the type of indices estimated or the characteristics of the model response, may emerge as more favorable from the perspective of computational efficiency and/or accuracy [6, 26].

When examining applications to models with multiple outputs, efforts to accommodate higher efficiency in the estimation of Sobol' indices [27-31] rely frequently on some form of output decomposition/compression or dimensionality reduction using Principal Component Analysis (PCA). The sensitivity in this case can be quantified with respect to the basis used to accomplish the dimensionality reduction (for example, the principal components when PCA is used), though, in many applications, transformation to the original output might be required to accommodate the insights needed of the GSA [29, 30]. As demonstrated in [29], PCA can facilitate an efficient, purely data-driven implementation in such a setting. In the [29] formulation, termed Dimension Reduction and Surrogate-based Sensitivity Analysis (DRE-SSA), PCA is first used to identify a low-dimensional space of latent components for the provided input/output samples, the necessary variance and covariance statistics are then estimated for the latent components, and finally the PCA linear mapping is utilized to obtain the Sobol' sensitivity indices for all original outputs. Since the statistics needed to accommodate the GSA correspond to the low dimensional latent space, they can be efficiently estimated individually. In [29] these statistics were approximated using a surrogate modeling approach. This paper offers a different formulation, considering the extension of PM-GSA [25] to accommodate the estimation of Sobol' indices within a setting that uses a PCA-based latent space representation. We should stress that the motivation of the present study is not any need to address shortcomings of the surrogate-model formulation in [29], which was shown to enjoy good accuracy and efficiency characteristics. Rather the proposed here advances merely establish an implementation that offers an alternative approach to the use of metamodeling techniques. As mentioned in the previous paragraph, such an alternative implementation can be proven useful for some application setups.

A data-driven framework is, therefore, established in this work, integrating the approaches proposed in [29] and [25], for the efficient estimation of Sobol' sensitivity indices for applications involving complex numerical models and high-dimensional outputs. The PCA-based formulation of Li et al. [29] is leveraged to restrict the estimation of statistics to the latent outputs only. The probability model-based GSA of Hu and Mahadevan [25] is then extended to estimate these statistics to replace the original surrogate-model formulation examined in [29]. The extension, constituting the main novel contribution of this work, refers to advances to accommodate the estimation of covariance statistics involving multiple latent outputs, beyond the estimation of variance statistics for single QoIs originally considered in [25]. Only the Gaussian mixture probability-based model investigated in [25] is considered here, due to the robust performance it exhibits, though the approach can be extended to the other ones examined in that paper. The impact on the accuracy of probability model-based GSA by the transformation of output space established through PCA is also examined in detail. The overall framework is termed PCA and Probability model-based sensitivity analysis (PCA-PSA).

The remainder of the paper is organized as follows. Section 2 reviews the problem formulation, and Section 3 develops the proposed method, offering also a concise review of the two methodologies that form its foundation. Section 4 considers two illustrative examples from the domain of natural hazards engineering, the first one considering the sensitivity of the engineering demand parameters (drifts and accelerations) of a 9-story benchmark steel structure with uncertain model properties exposed to seismic excitation (example with moderate output dimension), and the second one examining the sensitivity to storm forecast variability of the estimated peak-surge over an extended spatial grid (output with over million dimensions) during Superstorm Sandy. Across both examples, different settings are examined for the proposed formulation with respect to the number of latent outputs considered and the dimensionality of the GMM.

2. Problem formulation

Consider a system model with input vector $\mathbf{x}=[x_1,...,x_{n_x}]^T \in \mathbb{R}^{n_x}$, where x_i is the *i*th input and n_x is the total number of inputs, and let $y_j \in \mathbb{R}$ denote the *j*th model output (QoI). The system model has a total of n_y different outputs of interest, creating the output vector $\mathbf{y}=[y_1,...,y_{n_y}]^T \in \mathbb{R}^{n_y}$. Our underlying assumption is that the system model corresponds to a computationally expensive computer simulation, creating a restriction on the model evaluations that can be considered to perform the GSA (computational burden constraint). Note, though, that the established formulation is appropriate for any application with higher dimensional output vector. Let $f(\mathbf{x})$ denote the probability density function characterizing the variability of the system input. Although Sobol' indices are typically expressed for independent and uniformly distributed in [0,1] inputs, the concept has been readily extended to other types of distributions [32-34] as well as to jointly independent subsets of inputs [35]. The idea can be also implemented for individual dependent inputs, though in this case, care is needed in interpreting the exact meaning of the estimated sensitivity indices [36-38].

Variance-based GSA considers the decomposition of the total variance $Var[y_j]$ to the contributions coming from each of the inputs x_i as well as by the interaction between all groups of inputs [14]. Though Sobol’ indices of different orders of interaction can be defined, the two types of indices widely utilized are the first-order indices and the total-effect indices. For the *j*th output and the *i*th input, these indices are defined, respectively, as:

$$S_j^i = \frac{Var_i[E_{\sim i}[y_j \mid x_i]]}{Var[y_j]} = \frac{V_j^i}{V_j} \quad \forall i = 1,\ldots,n_x,\ j = 1,\ldots,n_y \tag{1}$$

$$S_j^{iT} = 1 - \frac{Var_{\sim i}[E_i[y_j \mid \mathbf{x}_{\sim i}]]}{Var[y_j]} = 1 - \frac{V_j^{\sim i}}{V_j} \quad \forall i = 1,\ldots,n_x,\ j = 1,\ldots,n_y \tag{2}$$

where Var[.] and E[.] denote the variance and expectation operators, respectively, $\mathbf{x}_{\sim i}$ denotes the input vector excluding x_i input, and subscripts i and $\sim i$ are utilized to describe statistics (variance or expectation) with respect to x_i or $\mathbf{x}_{\sim i}$ inputs, respectively. Also, for notational simplicity, and to accommodate the PCA decomposition examined in the next Section, we have defined $V_j = Var[y_j]$, $V_j^i = Var_i[E_{\sim i}[y_j \mid x_i]]$ and $V_j^{\sim i} = Var_{\sim i}[E_i[y_j \mid x_{\sim i}]]$. To facilitate a multi-output implementation for both the Sobol' indices and the associated statistics, the terminology established is to use subscripts for the output and superscripts for the input. The first-order indices given by Eq. (1) quantify the contribution of the ith input to the variability of the jth QoI without considering its interaction with the other inputs, while the total-effect indices given by Eq. (2) consider, additionally, the interactions with all possible combinations of the remaining inputs. Higher-order indices are also defined by considering interactions of multiple inputs. For example, the second-order sensitivity index between i and l inputs is given by:

$$S_j^{il} = \frac{Var_{il}[E_{\sim il}[y_j \mid x_i, x_l]] - V_j^i - V_j^l}{Var[y_j]} = \frac{V_j^{il} - V_j^i - V_j^l}{Var[y_j]} \quad \forall i,l = 1,\ldots,n_x,\ i \neq l,\ j = 1,\ldots,n_y \tag{3}$$

where we defined $V_j^{il} = Var_{il}[E_{\sim il}[y_j \mid x_i, x_l]]$.

A unified representation for the variances in the numerator of Eqs. (1), (2) and (3) can be established by defining:

$$V_j^{\mathbf{c}} = Var_{\mathbf{c}}[E_{\sim \mathbf{c}}[y_j \mid \mathbf{x}_{\mathbf{c}}]] \tag{4}$$

where **c** corresponds to the indices of inputs that need to be considered, with $\mathbf{x}_{\mathbf{c}}$ representing these inputs and $\mathbf{x}_{\sim \mathbf{c}}$ their complement. Dimension of $\mathbf{x}_{\mathbf{c}}$ will be denoted n_c herein. For Eq. (1) **c**=i, for Eq. (2) **c**=$\sim i$, while for Eq. (3) **c**=[i l]. A similar concept extends to the expressions for all other higher-order indices [12], including the estimation of first-order indices for subsets of inputs [35] for which **c** in Eq. (4) represents the subset index. Therefore, the calculation of the Sobol' indices requires estimation of V_j and estimation of $V_j^{\mathbf{c}}$ for different definitions for **c**.

Our objective in this paper is to estimate Sobol' indices for applications with a large number of outputs n_y, utilizing only a small number of evaluations of the system model (needed to estimate V_j and $V_j^{\mathbf{c}}$) to accommodate the assumed high computational complexity of the system model. The focus is on the estimation of the indices separately for each of the output components. Parenthetically note that this information can be subsequently used to define aggregated (across all outputs) importance indicators for each input definition $\mathbf{x}_\mathbf{c}$. This is accomplished by a weighted average of the sensitivity for each output *j*, with the variance V_j as recommended weight [22, 39]. Within the context of the PCA dimensionality reduction approach, implemented in the next section, an equivalent derivation of aggregated importance indicators can be accomplished by utilizing the sensitivity for each principal component (instead of the sensitivity for each of the original outputs) in this formulation.

3. Proposed method

As discussed in the introduction, an entirely data-driven formulation is examined for the Sobol' indices estimation. Assume that a total of *k* model evaluations (simulations) is available, for different samples for the model input $\{\mathbf{x}^s; s=1,...,k\}$. These samples are obtained from the underlying probability distribution *f*(**x**). The corresponding output is $\{\mathbf{y}^s; s=1,...,k\}$ with $\mathbf{y}^s = \mathbf{y}(\mathbf{x}^s)$ representing the output vector for model input $\mathbf{x}^s$. Let finally $\mathbf{X} = [\mathbf{x}^1 \ ... \ \mathbf{x}^k]^T \in \mathbb{R}^{k \times n_x}$ and $\mathbf{Y} = [\mathbf{y}^1 \ ... \ \mathbf{y}^k]^T \in \mathbb{R}^{k \times n_y}$ denote the input and output matrices, respectively. Our objective is to estimate Sobol' indices using data [**X**, **Y**]. Sections 3.1 and 3.2 discuss the two components of the proposed PCA-PSA framework, the dimensionality reduction and probability model-based estimation of relevant statistics, respectively, while Section 3.3 reviews the overall implementation and discusses computational efficiency and accuracy characteristics.

3.1 Dimensionality reduction using PCA

PCA is first used as a dimensionality reduction technique [40]. In this setting, it is convenient to consider PCA as the eigendecomposition of the covariance matrix $\bar{\mathbf{Y}}^T\bar{\mathbf{Y}}$ associated with the observation matrix $\mathbf{Y}$, where $\bar{\mathbf{Y}}$ corresponds to the matrix of normalized observations, established by subtracting for each output its mean value over the observations:

$$\begin{aligned} \bar{y}_j &= y_j - \frac{1}{k}\sum_{s=1}^{k} y_j^s = y_j - \mu_{y_j} \\ \text{where } \mu_{y_j} &= \frac{1}{k}\sum_{s=1}^{k} y_j^s \end{aligned} \tag{5}$$

Solution of the eigenvalue problem provides the vector of latent outputs (also mentioned as principal components) with the *j*th latent output denoted by z_j. The corresponding eigenvalue λ_j represents the portion of the total variance of the original data **Y** that can be explained by z_j, while the corresponding eigenvector $\mathbf{P}_j \in \mathbb{R}^{n_y}$ facilitates the mapping from **y** to z_j. The number of such independent latent outputs equals to the rank of $\bar{\mathbf{Y}}$ [40], corresponding to min(n_y,*k*-1) [minimum of the number of independent rows or columns], where min(a,b) denotes the minimum of the two arguments. To accommodate a larger dimensionality reduction only the principal components corresponding to the n_p largest eigenvalues are retained, with n_p chosen so that the ratio

$$r = \frac{\sum_{j=1}^{n_p} \lambda_j}{\sum_{j=1}^{\min(n_y,k-1)} \lambda_j} \tag{6}$$

is greater than some threshold r_o (for example 99.9%). This ratio represents the portion of the original output variance that can be explained by the retained components [41]. In Eq. (6), the denominator represents the total variance, while the numerator the variance of the retained components. The selection of n_p will be further examined within the context of the overall framework in Section 3.3. The vector of the retained principal components is denoted by **z**, while the relationship between **z** and **y** is $\mathbf{y} = \mathbf{P}\mathbf{z} + \boldsymbol{\mu}_y + \boldsymbol{\tau}$ where $\mathbf{P} \in \mathbb{R}^{n_y \times n_p}$ is the projection matrix with the jth column corresponding to eigenvectors $\mathbf{P}_j$, $\boldsymbol{\mu}_y$ is the mean vector for the original data **Y** (vector with elements μ_{y_j}) and $\boldsymbol{\tau}$ represents the PCA approximation error. This means that for the jth original output the following approximation is established:

$$y_j \approx [\mathbf{P}]_{j*}\mathbf{z} + \mu_{y_j} \tag{7}$$

where $[\mathbf{P}]_{j*} \in \mathbb{R}^{1\times n_p}$ is the row vector corresponding to the jth row of the projection matrix **P**. The observation matrix $\mathbf{Z} = [\mathbf{z}^1 \ ... \ \mathbf{z}^k]^T \in \mathbb{R}^{k\times n_p}$ for the latent outputs is $\mathbf{Z} = \overline{\mathbf{Y}}\mathbf{P}$, with sth row corresponding to the latent output vector for model input $\mathbf{x}^s$.

Using information [**X**, **Z**] calculation of statistics of interest for the latent output **z**, to support the estimation of the desired sensitivity indices for **y**, will be discussed in Section 3.2. Specifically, calculation of V_j requires the estimation of the covariance matrix $\boldsymbol{\Sigma}_\mathbf{z}$ for **z**, and calculation of $V_j^\mathbf{c}$ requires estimation of the covariance matrix $\boldsymbol{\Sigma}_\mathbf{z}^\mathbf{c}$ for random variables $E_{\sim\mathbf{c}}[z_j \,|\, \mathbf{x}_\mathbf{c}]$ [29]. Note that the diagonal elements of $\boldsymbol{\Sigma}_\mathbf{z}^\mathbf{c}$ correspond to $Var_\mathbf{c}[E_{\sim\mathbf{c}}[z_j \,|\, \mathbf{x}_\mathbf{c}]]$ while the jl off-diagonal element corresponds to $Cov_\mathbf{c}[E_{\sim\mathbf{c}}[z_j \,|\, \mathbf{x}_\mathbf{c}], E_{\sim\mathbf{c}}[z_l \,|\, \mathbf{x}_\mathbf{c}]]$ where Cov[.] corresponds to the covariance operator.

Based on transformation of Eq. (7), and given statistics $\boldsymbol{\Sigma}_\mathbf{z}$, and $\boldsymbol{\Sigma}_\mathbf{z}^\mathbf{c}$ for **z**, the quantities needed for the estimation of sensitivity indices for each of the original outputs can be approximated as [29]:

$$\begin{aligned} V_j &= [\mathbf{P}]_{j*}\boldsymbol{\Sigma}_\mathbf{z}([\mathbf{P}]_{j*})^T \\ V_j^\mathbf{c} &= [\mathbf{P}]_{j*}\boldsymbol{\Sigma}_\mathbf{z}^\mathbf{c}([\mathbf{P}]_{j*})^T \end{aligned} \tag{8}$$

For the entire output vector **y**, these statistics can be conveniently expressed in matrix form [29]. For example, for $V_j^{\mathbf{c}}$ we have:

$$\mathbf{V}^{\mathbf{c}} = \text{diag}(\mathbf{P}\boldsymbol{\Sigma}_{\mathbf{z}}^{\mathbf{c}}(\mathbf{P})^T) = \sum_{row}\left([\mathbf{P}\boldsymbol{\Sigma}_{\mathbf{z}}^{\mathbf{c}}] \circ \mathbf{P}\right) \tag{9}$$

where $\mathbf{V}^{\mathbf{c}} \in \mathbb{R}^{n_y}$ denotes the vector with elements $V_j^{\mathbf{c}}$, diag(**A**) corresponds to the diagonal elements of matrix **A**, $\mathbf{A} \circ \mathbf{B}$ corresponds to the Hadamard product of matrices **A** and **B** (element-wise multiplication of the matrices), and $\sum_{row}\mathbf{A}$ represents the vector with elements given by the summation of the matrix **A** elements for each column (summation across the matrix rows). Identical expression holds for V_j:

$$\mathbf{V} = \text{diag}(\mathbf{P}\boldsymbol{\Sigma}_{\mathbf{z}}(\mathbf{P})^T) = \sum_{row}\left([\mathbf{P}\boldsymbol{\Sigma}_{\mathbf{z}}] \circ \mathbf{P}\right) \tag{10}$$

where $\mathbf{V} \in \mathbb{R}^{n_y}$ denotes the vector with elements V_j. Note that statistics V_j can be easily derived from matrix **Y**, though to establish a consistent influence of the PCA truncation error across all statistics needed for the Sobol' index estimation, it is recommended to use the latent output statistics also to calculate V_j. This way, the truncation error originating from the reduced number of components retained in the PCA impacts all the calculated statistics for y_j.

Finally, according to the decomposition of Eq. (8) [or of the equivalent decomposition in vector format given by Eq. (9)], estimation of the Sobol' indices for the original output requires estimation of matrices $\boldsymbol{\Sigma}_{\mathbf{z}}$ and $\boldsymbol{\Sigma}_{\mathbf{z}}^{\mathbf{c}}$, which needs to be performed given the data [**X**, **Z**]. The first one is given by $\boldsymbol{\Sigma}_{\mathbf{z}} = \mathbf{Z}^T\mathbf{Z}/(k-1)$, while the estimation of the latter is discussed in the next Section.

3.2 Probability model-based estimation of conditional statistics

The estimation of the conditional statistics $\boldsymbol{\Sigma}_{\mathbf{z}}^{\mathbf{c}}$ using data [**X**, **Z**] is established utilizing the probability model-based approach of Hu and Mahadevan [25]. As discussed in the introduction, a Gaussian mixture is adopted here as the probability model, though formulation can be readily extended to accommodate the copula-based probability models examined in detail in [25]. The approximations for the variance-related statistics $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}]]$ are established following directly the formulation in [25], while the approximations of the covariance statistics $Cov_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}], E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}]]$ require a slight extension. The variance-related statistics are reviewed first, followed by the extension to the covariance statistics. A slight modification of the Monte Carlo integration needed for the variance statistics is also introduced.

For estimating $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}]]$ the joint probability density function (PDF) of $\mathbf{x}_{\mathbf{c}}$ and z_j (n_c+1 dimensional PDF) is initially approximated through a multivariate GMM utilizing subset $[\mathbf{X}_{\mathbf{c}}, \mathbf{Z}_j]$ of the original data, where $\mathbf{X}_{\mathbf{c}}$ corresponds to data for input components $\mathbf{x}_{\mathbf{c}}$ and $\mathbf{Z}_j$ corresponds to the data for principal component z_j. Note that this projection of the data to the space of $[\mathbf{x}_{\mathbf{c}}, z_j]$ ultimately accommodates the estimation of statistics with respect to $\mathbf{x}_{\sim\mathbf{c}}$ [25]. The approximation of the desired joint PDF is written as:

$$f(\mathbf{x}_{\mathbf{c}}, z_j) = \sum_{q=1}^{Q} w_q^{(j)} \mathcal{N}(\mathbf{x}_{\mathbf{c}}, z_j \mid \boldsymbol{\mu}_q^{(j)}, \boldsymbol{\Sigma}_q^{(j)}) \tag{11}$$

where Q is the number of GMM components, $w_q^{(j)}$, $\boldsymbol{\mu}_q^{(j)}$, and $\boldsymbol{\Sigma}_q^{(j)}$ are the weight, mean vector and covariance matrix of the qth GMM component, respectively, and $\mathcal{N}(\mathbf{v}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes Gaussian PDF with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ evaluated at $\mathbf{v}$. Superscript (j) is used for all relevant GMM parameters to distinguish that the GMM fit pertains to $\mathbf{x}_{\mathbf{c}}$ and z_j combination. The mean and covariance of the qth component can be partitioned as:

$$\begin{aligned} \boldsymbol{\mu}_q^{(j)} &= \begin{bmatrix} \boldsymbol{\mu}_{q,\mathbf{x}_{\mathbf{c}}}^{(j)} \\ \mu_{q,z_j}^{(j)} \end{bmatrix} \\ \boldsymbol{\Sigma}_q^{(j)} &= \begin{bmatrix} \boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}}^{(j)} & \boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}z_j}^{(j)} \\ \boldsymbol{\Sigma}_{q,z_j\mathbf{x}_{\mathbf{c}}}^{(j)} & \boldsymbol{\Sigma}_{q,z_j}^{(j)} \end{bmatrix} \end{aligned} \tag{12}$$

where subscript $\mathbf{x}_{\mathbf{c}}$ or z_j (or combined) is used to denote the specific random variable of the joint PDF the statistic pertains to. Note that $\boldsymbol{\Sigma}_{q,z_j}^{(j)}$ is a scalar (the variance of z_j for the GMM fit) but for notational consistency, it is denoted as a covariance matrix. The GMM parameters can be estimated using the Expectation Maximization (EM) algorithm [42], whereas for selecting the number of components, techniques relying on the Bayesian Information Criterion can be utilized to avoid overfitting that data [43]. Based on the fitted GMM of Eq. (11), the conditional expectation $E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}]$ can be expressed using the conditional distribution of the fitted Gaussian mixture probability model as [25]:

$$E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}] = \sum_{q=1}^{Q} w_q^{(j)}(\mathbf{x}_{\mathbf{c}}) \cdot \mu_{q,z_j|\mathbf{x}_{\mathbf{c}}}^{(j)} \tag{13}$$

where conditional statistics for each component of the GMM are obtained as:

$$\begin{aligned} w_q^{(j)}(\mathbf{x}_{\mathbf{c}}) &= \frac{w_q^{(j)} \mathcal{N}(\mathbf{x}_{\mathbf{c}} \mid \boldsymbol{\mu}_{q,\mathbf{x}_{\mathbf{c}}}^{(j)}, \boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}}^{(j)})}{\sum_{q'=1}^{Q} w_{q'}^{(j)} \mathcal{N}(\mathbf{x}_{\mathbf{c}} \mid \boldsymbol{\mu}_{q',\mathbf{x}_{\mathbf{c}}}^{(j)}, \boldsymbol{\Sigma}_{q',\mathbf{x}_{\mathbf{c}}}^{(j)})} \\ \mu_{q,z_j|\mathbf{x}_{\mathbf{c}}}^{(j)} &= \mu_{q,z_j}^{(j)} + \boldsymbol{\Sigma}_{q,z_j\mathbf{x}_{\mathbf{c}}}^{(j)} \left(\boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}}^{(j)}\right)^{-1} (\mathbf{x}_{\mathbf{c}} - \boldsymbol{\mu}_{q,\mathbf{x}_{\mathbf{c}}}^{(j)}) \end{aligned} \tag{14}$$

These expressions can be obtained [25, 44] by leveraging the linearity of the expectation operator [Eq. (13)], the marginal distribution for $\mathbf{x_c}$ [appearing in the denominator of the weights $w_q^{(j)}(\mathbf{x_c})$], and the conditional expectation of each of the Gaussian mixture components [appearing in the expression for $\mu_{q,z_j|\mathbf{x_c}}^{(j)}$]. The overall variance $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]]$ can be finally calculated using Monte Carlo Integration (MCI) to estimate the variance of the conditional expectation, with the latter approximated through Eq. (13). The resultant expression is:

$$Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]] \approx \frac{1}{N}\sum_{h=1}^{N}\left\{E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}^{(h)}] - \left(\frac{1}{N}\sum_{h=1}^{N}E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}^{(h)}]\right)\right\}^2 \tag{15}$$

where N is the total number of MC samples used and $\mathbf{x_c}^{(h)}$ denotes the hth sample of $\mathbf{x_c}$. There are two choices for generating the sample set $\{\mathbf{x_c}^{(h)}; h = 1,...,N\}$. The first one, denoted as MCI$_f$, is to obtain samples from $f(\mathbf{x_c})$ [for example using directly the sample set $\mathbf{X_c}$], while the second one, the one recommended in [25] and denoted as MCI$_p$, is to utilize instead of $f(\mathbf{x_c})$ the marginalized fitted GMM PDF of Eq. (11) for $\mathbf{x_c}$ given by:

$$\mathbf{x_c} \sim \sum_{q=1}^{Q} w_q^{(j)} \mathcal{N}(\mathbf{x_c} \mid \boldsymbol{\mu}_{q,\mathbf{x_c}}^{(j)}, \boldsymbol{\Sigma}_{q,\mathbf{x_c}}^{(j)}) \tag{16}$$

This PDF corresponds to the approximation of $f(\mathbf{x_c})$ based on the GMM fitted on [$\mathbf{x_c}$ z_j]. The objective behind using the fitted approximation instead of the original PDF is to accommodate consistency: both the conditional expectation $E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]$ as well as the overall variance $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]]$ are approximated using the fitted probability model. Independent of the approach adopted to generate the set $\{\mathbf{x_c}^{(h)}; h = 1,...,N\}$, the computational burden of the MCI is small, therefore value for N can be selected large to minimize the estimation error.

The procedure for approximating the covariance statistic $Cov_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}], E_{\sim\mathbf{c}}[z_l \mid \mathbf{x_c}]]$ is similar and it involves two steps: approximation of the two conditional expectations using the conditional GMM and numerical integration for the covariance statistics by MCI. Depending on the approach taken for generating the sample set $\{\mathbf{x_c}^{(h)}; h = 1,...,N\}$ for the MCI, the required conditional expectations might be readily available through existing results (for calculating variance statistics) or they might need new GMM fits. If the sample set $\{\mathbf{x_c}^{(h)}; h = 1,...,N\}$ is obtained from $f(\mathbf{x_c})$ (MCI$_f$) then the two conditional expectations $E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]$ and $E_{\sim\mathbf{c}}[z_l \mid \mathbf{x_c}]$ are readily available from the GMM established to estimate $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]]$ (using data [$\mathbf{X_c}$, $\mathbf{Z}_j$]) and $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_l \mid \mathbf{x_c}]]$ (using data [$\mathbf{X_c}$, $\mathbf{Z}_l$]), respectively, expressed through Eqs. (13) and (14). The MCI estimate of the covariance statistics is then:

$$
\begin{aligned}
&Cov_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}],\ E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}]] \\
&\quad \approx \frac{1}{N}\sum_{h=1}^{N}\left\{E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}^{(h)}]-\left(\frac{1}{N}\sum_{h=1}^{N}E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}^{(h)}]\right)\right\}\left\{E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}^{(h)}]-\left(\frac{1}{N}\sum_{h=1}^{N}E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}^{(h)}]\right)\right\} \\
&\quad \approx \frac{1}{N}\sum_{h=1}^{N}E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}^{(h)}]E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}^{(h)}]-\left(\frac{1}{N}\sum_{h=1}^{N}E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}^{(h)}]\right)\left(\frac{1}{N}\sum_{h=1}^{N}E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}^{(h)}]\right)
\end{aligned}
\tag{17}
$$

If, on the other hand, the sample set $\{\mathbf{x}_{\mathbf{c}}^{(h)}; h=1,...,N\}$ utilized in the MCI needs to come from the same GMM that is used for estimation of the conditiona l statistics (MCI$_p$), then a new probability model is needed. For this purpose, the joint PDF of $\mathbf{x}_{\mathbf{c}}$ and z_j and z_l (n_c+2 dimensional PDF) is initially approximated through a GMM utilizing subset $[\mathbf{X}_{\mathbf{c}},\ \mathbf{Z}_j,\ \mathbf{Z}_l]$ of the original data. The corresponding PDF is:

$$
f(\mathbf{x}_{\mathbf{c}}, z_j, z_l)=\sum_{q=1}^{Q} w_q^{(j,l)} \mathcal{N}(\mathbf{x}_{\mathbf{c}}, z_j, z_l \mid \boldsymbol{\mu}_q^{(j,l)}, \boldsymbol{\Sigma}_q^{(j,l)})
\tag{18}
$$

where the mean and covariance of each component can be partitioned as:

$$
\begin{gathered}
\boldsymbol{\mu}_q^{(j,l)}=\begin{bmatrix}\boldsymbol{\mu}_{q,\mathbf{x}_{\mathbf{c}}}^{(j,l)} \\ \mu_{q,z_j}^{(j,l)} \\ \mu_{q,z_l}^{(j,l)}\end{bmatrix} \\
\boldsymbol{\Sigma}_q^{(j,l)}=\begin{bmatrix}\boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}}^{(j,l)} & \boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}z_j}^{(j,l)} & \boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}z_l}^{(j,l)} \\ \boldsymbol{\Sigma}_{q,z_j\mathbf{x}_{\mathbf{c}}}^{(j,l)} & \boldsymbol{\Sigma}_{q,z_j}^{(j,l)} & \boldsymbol{\Sigma}_{q,z_jz_l}^{(j,l)} \\ \boldsymbol{\Sigma}_{q,z_l\mathbf{x}_{\mathbf{c}}}^{(j,l)} & \boldsymbol{\Sigma}_{q,z_lz_j}^{(j,l)} & \boldsymbol{\Sigma}_{q,z_l}^{(j,l)}\end{bmatrix}
\end{gathered}
\tag{19}
$$

Superscript (j,l) is used to denote the fact that the GMM fit pertains to the combination of $\mathbf{x}_{\mathbf{c}}$, z_j and z_l. Conditional expectations $E_{\sim\mathbf{c}}[z_j \mid \mathbf{x}_{\mathbf{c}}]$ and $E_{\sim\mathbf{c}}[z_l \mid \mathbf{x}_{\mathbf{c}}]$ in this instance are obtained by first estimating the marginal distribution with respect to the desired variables, $[\mathbf{x}_{\mathbf{c}}\ z_j]$ and $[\mathbf{x}_{\mathbf{c}}\ z_l]$, respectively, and then by calculating the conditional statistics with respect to $\mathbf{x}_{\mathbf{c}}$, which leads to:

$$
\begin{gathered}
E_{\sim\mathbf{c}}[z_o \mid \mathbf{x}_{\mathbf{c}}]=\sum_{q=1}^{Q} w_q^{(j,l)}(\mathbf{x}_{\mathbf{c}})\cdot \mu_{q,z_o|\mathbf{x}_{\mathbf{c}}}^{(j,l)} \quad ;\ o=j \text{ or } l \\
\text{where } w_q^{(j,l)}(\mathbf{x}_{\mathbf{c}})=\frac{w_q^{(j,l)}\mathcal{N}(\mathbf{x}_{\mathbf{c}} \mid \boldsymbol{\mu}_{q,\mathbf{x}_{\mathbf{c}}}^{(j,l)},\ \boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}}^{(j,l)})}{\sum_{q'=1}^{Q} w_{q'}^{(j,l)}\mathcal{N}(\mathbf{x}_{\mathbf{c}} \mid \boldsymbol{\mu}_{q',\mathbf{x}_{\mathbf{c}}}^{(j,l)},\ \boldsymbol{\Sigma}_{q',\mathbf{x}_{\mathbf{c}}}^{(j,l)})} \\
\mu_{q,z_o|\mathbf{x}_{\mathbf{c}}}^{(j,l)}=\mu_{q,z_o}^{(j,l)}+\boldsymbol{\Sigma}_{q,z_o\mathbf{x}_{\mathbf{c}}}^{(j,l)}\left(\boldsymbol{\Sigma}_{q,\mathbf{x}_{\mathbf{c}}}^{(j,l)}\right)^{-1}(\mathbf{x}_{\mathbf{c}}-\boldsymbol{\mu}_{q,\mathbf{x}_{\mathbf{c}}}^{(j,l)})
\end{gathered}
\tag{20}
$$

Covariance statistics are finally obtained by Eq. (17) with conditional expectations provided by Eq. (20) and the sample set $\{\mathbf{x}_{\mathbf{c}}^{(h)}; h=1,...,N\}$ obtained through the marginalized fitted GMM PDF of Eq. (18) for $\mathbf{x}_{\mathbf{c}}$ given in this case by:

$$\mathbf{x_c} \sim \sum_{q=1}^{Q} w_q^{(j,l)} \mathcal{N}(\mathbf{x_c} \mid \boldsymbol{\mu}_{q,\mathbf{x_c}}^{(j,l)}, \boldsymbol{\Sigma}_{q,\mathbf{x_c}}^{(j,l)}) \tag{21}$$

Finally, the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{z}}^{\mathbf{c}}$ is assembled by using diagonal components $Var_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}]]$ and off-diagonal components $Cov_{\mathbf{c}}[E_{\sim\mathbf{c}}[z_j \mid \mathbf{x_c}], E_{\sim\mathbf{c}}[z_l \mid \mathbf{x_c}]]$.

3.3 Overview of PCA-PSA algorithm and discussion on computational characteristics

Combining the concepts discussed in the previous two sections, the PCA and Probability model-based sensitivity analysis (PCA-PSA) algorithm is established through the following steps.

Step 1: Generate sample set $\{\mathbf{x}^s; s=1,...,k\}$ ~$f(\mathbf{x})$ and evaluate system output $\{\mathbf{y}^s; s=1,...,k\}$ for all of them to obtain data [**X**, **Y**] (with rows corresponding to the input and output vectors, respectively, for each sample).

Step 2: Perform principal component analysis and identify eigenvalues λ_j and eigenvectors $\mathbf{P}_j$ of covariance matrix $\overline{\mathbf{Y}}^T\overline{\mathbf{Y}}$. Retain the principal components that correspond to the n_p largest eigenvalues so that the ratio r given by Eq. (6) is greater than desired threshold r_o (for example 99% or 99.9%). This provides the projection matrix **P** (with columns the retained eigenvectors) and the data matrix for the principal components $\mathbf{Z} = \overline{\mathbf{Y}}\mathbf{P}$. Estimate the variance of the original output using the retained principal components through Eq. (10) using $\boldsymbol{\Sigma}_{\mathbf{z}} = \mathbf{Z}^T\mathbf{Z}/(k-1)$**.**

For each of the Sobol' indices perform the following steps. These steps are presented next for the MCI$_f$ implementation, with extension to MCI$_p$ examined later.

Step 3: Based on the calculated index define subset $\mathbf{x_c}$. This also determines its complement $\mathbf{x}_{\sim\mathbf{c}}$ but that complement will not be explicitly needed.

Step 4: For each principal component z_j, fit a GMM to data [$\mathbf{X_c}$, $\mathbf{Z}_j$] where $\mathbf{X_c}$ corresponds to the columns of **X** corresponding to $\mathbf{x_c}$ and $\mathbf{Z}_j$ to the jth column of **Z**. Repeat this n_p times, one for each z_j.

Step 5: Estimate the variance statistics for each z_j using Eq. (15) and covariance statistics using Eq. (17) with sample set $\{\mathbf{x}_{\mathbf{c}}^{(h)}; h=1,...,N\}$ obtained from $f(\mathbf{x_c})$ and all conditional expectations estimated according to Eqs. (13) and (14) using the GMM for the respective output z_j.

Step 6: Calculate the required variance statistics for the original output using Eq. (9) and use that to estimate the Sobol' indices for the original output.

If MCI$_p$ implementation is utilized, then Steps 4 and 5 need to be modified. In this case, each variance or covariance statistic is based on a separate GMM fit. Dedicating each of these two steps to the estimation of the respective statistics, the alternative formulation has the following modified steps:

Step 4: For each principal component z_j, fit a GMM to data [$\mathbf{X_c}$, $\mathbf{Z}_j$] and estimate the variance statistics using Eq. (15) with sample set $\{\mathbf{x}_\mathbf{c}^{(h)}; h=1,...,N\}$ obtained from Eq. (16), and conditional expectation estimated according to Eqs. (13) and (14). Repeat this n_p times, one for each z_j.

Step 5: For each combination of principal components z_j and z_l, $l \neq j$, fit a GMM to data [$\mathbf{X_c}$, $\mathbf{Z}_j$, $\mathbf{Z}_l$] and estimate the covariance statistics using Eq. (17) with sample set $\{\mathbf{x}_\mathbf{c}^{(h)}; h=1,...,N\}$ obtained from Eq. (21), and conditional expectations estimated according to Eq. (20). Repeat this $n_p(n_p$-1)/2 times, one for each combination of z_j and $z_{l.}$

The computationally intensive part of the PCA-PSA algorithm is the GMM fit to the data, which for each index needs to be performed n_p times if MCI$_f$ is adopted and $n_p(n_p$+1)/2 times if MCI$_p$ is adopted. For the MCI$_f$ implementation the fit pertains to a n_c+1 dimensional PDF and for the MCI$_p$ implementation to a n_c+1 dimensional or a n_c+2 dimensional PDF. Most other calculations rely on simple matrix manipulations, and so can be performed with a negligible computational burden. The Monte Carlo integration also has a very small burden even for large N values, since the quantities involved (conditional expectations) have only small computational complexity. This burden is reduced in the MCI$_f$ implementation, since for the covariance statistics the same conditional expectation calculations as for the variance statistics can be utilized. On the other hand, for the MCI$_p$ implementation, the estimation of covariance statistics requires separate calculations for the conditional expectation approximations. Finally, the computational burden for the PCA can be moderate if n_y and k are large, but the PCA needs to be performed only once; therefore, the contribution to the total computational burden is small. Thus, the overall computational complexity of the PCA-PSA algorithm primarily comes from the GMM fit. This shows that MCI$_f$ provides substantial computational advantages compared to MCI$_p$.

Furthermore, when compared to the implementation without dimensionality reduction, which would require n_y different GMM fits, it is evident that the proposed formulation can give substantial benefits when n_p<<n_y. A further improvement of computational efficiency for PCA-PSA can be accomplished if a single GMM is considered for all principal components. This is accomplished by considering a GMM fit for the joint PDF of $\mathbf{x_c}$ and $\mathbf{z}$ in Step 4 of the original algorithm, using data [$\mathbf{X_c}$, $\mathbf{Z}$]. Then for the estimation of the conditional statistics, a marginalization of the remaining outputs is first performed to obtain the GMM fit for $\mathbf{x_c}$ and z_j (retain only the components of the mean vector and covariance matrix related to $\mathbf{x_c}$ and z_j) and that GMM is then utilized for estimating the necessary conditional statistics. This implementation reduces the number of fitted GMMs to 1, though for a n_c+n_p dimensional PDF of substantially larger dimension, and will be referenced as PCA-PSA$_s$ herein. Technically either MCI$_f$ or MCI$_p$ integration can be combined with this formulation.

Regarding the accuracy of PCA-PSA algorithm, and assuming that the Monte Carlo integration can be easily performed for a large value of *N* (therefore reducing the integration error), there are two main sources of error: (a) the error stemming from the dimensionality reduction due to PCA, and (b) the error stemming from the approximation of the conditional statistics using the GMM fit. The first type of error can be easily reduced by adopting a large value for the r_o threshold and keeping a large number of principal components [29]. This can be easily accommodated within PCA-PSA, especially if MCI_f is adopted since the computational burden increases only linearly with the number of principal components. In such cases, it is recommended that r_o is set to a very high value, over 99%. The influence of the second type of error is discussed in detail in [25], and fundamentally it depends on the quality of the probability model approximation established through the GMM. This error can be reduced by using a larger database (increasing *k*), but this might be impractical for applications with expensive simulation models. Also, even for larger *k* values, there will be an unavoidable error, associated with how well the adopted probability model (GMM in this case) fits the original response distribution. Finally, it is important to note that this error is impacted by the dimensionality of vector $\mathbf{x_c}$, which dictates the dimension for the fitted GMM (n_c+1 or n_c+2 dimensional PDF for the PCA-PSA implementation), and, therefore, is expected to increase for higher-order indices, and for total-effect indices [25]. For the same reasons, it will be larger for first-order indices for group of inputs, since in this case n_c corresponds to the size of the group.

The accuracy of PCA-PSA could be additionally impacted by the transformation of the output, i.e. the fact that the GMM fit and approximation of conditional distribution are implemented for each of the latent outputs instead of the original output. A priori there is no indication that one output space will always accommodate a better fit than the other one (this will be at best application-dependent), and so there should be no strong influence on the results by the output transformation itself. Overall, provided that a sufficient number of latent components is used, the accuracy of PCA-PSA is expected to be similar to the one established by the implementation of PM-GSA on the original output data, offering though substantial computational benefits for applications with large dimensional outputs. As will be shown in one of the examples the transformation of the output space may even act beneficially if the original output has some complex behavior.

Finally, some remarks should be provided for comparing PCA-PSA to the original surrogate-based DRE-SSA [29]. PCA-PSA ultimately replaces the surrogate model approximation for the latent response with respect to the entire input **x** required in DRE-SSA, with a GMM fit with respect to the same response and vector $\mathbf{x_c}$, which though needs to be repeated for each $\mathbf{x_c}$ definition. Both implementations need to be established for each latent response output (no differences with respect to this aspect). Although the GMM fit is expected to be a computationally simpler task, especially when the number of simulations *k* is larger, the fact that it needs to be repeated for each $\mathbf{x_c}$ definition might create a similar, or an even higher computational burden, depending on the number of different indices examined. With respect to the accuracy of each of these two approximations (GMM fit versus surrogate model approach), relative preference for each will depend on the specifics of the application considered [25, 26]. This issue, though, is independent of the output dimensionality, and so is outside the scope of the present study, which, as discussed in the introduction, aims to offer an alternative to the original surrogate-based DRE-SSA [29] formulation. For this reason, the comparisons in the illustrative examples next, will focus on aspects relevant to the higher dimensional output (comparing PCA-PSA to the version without the PCA implementation, PM-GSA) and not on the type of approximation (GMM fit or surrogate model-based DRE-SSA [29]) established.

4. Illustrative examples

Two different illustrative examples are considered, originating from the domain of natural hazards engineering. The first example examines the sensitivity of engineering demand parameters (drifts and accelerations) of a 9-story benchmark steel structure exposed to a seismic excitation and represents an example with a moderate output dimension (n_y=18), while the second example examines the sensitivity to storm forecast variability of the estimated peak-surge over an extended spatial grid (output with over million dimensions) during Superstorm Sandy. Emphasis is placed on the estimation of first-order indices though some results will be presented for higher-order indices and total-effect indices. Validation of PCA-PSA is established by comparing predictions to the exact results, obtained through a double-loop Monte Carlo integration. For first-order and total-effect indices, an efficient numerical implementation of the double-loop MCI is considered [14, 15]. For higher-order interaction indices, the double-loop MCI is separately implemented for each index. For both examples, the GMM fit is performed using the Expectation-Maximization algorithm [44] with an adaptive selection of the number of mixture components based on the maximization of the Bayesian Information Criterion [43].

For both examples, estimation of Sobol' indices using both PCA-PSA as well as directly for the original output by PM-GSA (no PCA step) is examined. To investigate the accuracy across the different variants for PCA-PSA, both MCI$_f$ and MCI$_p$ approaches are considered, whereas the PCA-PSA$_s$ formulation, implemented using MCI$_f$ integration, is also examined. For the reasons discussed in Section 3.3, no comparisons are performed with respect to the surrogate-based DRE-SSA [29] formulation.

Results will also be presented for individual outputs, but to allow for easier exploration of trends across different implementation variants the focus will be on average accuracy across the outputs using the normalized root mean squared error as a metric. If S_j denotes the reference results for the Sobol' index of interest for output y_j obtained through the double loop MCI and $\tilde{S}_j$ the approximation of the same index using PCA-PSA, the normalized root mean squared error, denoted *nrmse* herein, is given by:

$$nrmse = \frac{\sqrt{\frac{1}{n_y}\sum_{j=1}^{n_y}(S_j - \tilde{S}_j)^2}}{\max_{j=1,\ldots,n_y}(S_j) - \min_{j=1,\ldots,n_y}(S_j)} \tag{22}$$

4.1 Nine-story benchmark building under earthquake excitation

The first example corresponds to a nine-story, hysteretic moment resisting frame steel structure [45]. The numerical model for it is described in detail in [46], and it corresponds to a nonlinear, hysteretic model. The structure is excited by the Loma Prieta earthquake and its response is estimated through nonlinear response-history analysis. Output **y** includes peak values of the inter-story drifts of all stories $\{\delta_j; j=1, \ldots,9\}$ and absolute floor acceleration for all stories $\{a_j; j=1, \ldots,9\}$ for a total of n_y=18 outputs. Each of the two output types (drifts and accelerations) is normalized so that the contribution to the total variance is equal; this is done merely to establish a balanced contribution to the overall PCA from drifts and accelerations, and does not affect the GSA in any other way. It can be simply viewed as scaling of the drifts and accelerations through some proper thresholds for each engineering demand parameter type. The model input **x** includes the damping ratio, ζ, the modulus of elasticity, E, and yield stress, f_y, for the steel, and the mass coefficient (representing mass density per floor), m_s, for a total of n_x=4 inputs. For each input, a range of possible values is considered, expressed as variation with respect to the nominal values reported in [46]. This range is chosen as [0.8 1.2] for the modulus of elasticity, and [0.6 1.4] for the remaining parameters. The smaller range for the modulus of elasticity was chosen (after some initial investigation) so that it does not dominate the sensitivity of the output. A uniform probability distribution is chosen for input **x** within the aforementioned ranges. To improve the accuracy of the GMM fit, the transformation of **x** to the standard Gaussian space is performed when establishing that fit. Three different values for the number of total simulations will be examined, k=500, k=1000 and k=10,000, for the proposed data-driven GSA implementation. The first two should be considered reasonable values for seismic risk assessment for applications with complex numerical models, while the third, larger value is examined to better investigate the impact on the accuracy when larger amount of data is available.

Figure 1 presents the ratio of captured variance as a function of the number of retained principal components for implementations with k=500 or 10,000 simulations (model evaluations). Based on the results of the figure, for r_o=99%, n_p=10 number of components are needed whereas for r_o=99.9%, n_p=16 number of components are needed, practically independent of the value of k. Figure 2 shows the accuracy improvement for the estimation of the first-order Sobol' indices, quantified through *nrmse* value, as the number of retained principal components (n_p) increases for the case with k=500 simulations. Figure 3 shows the same results for the case with k=10,000 simulations. Note that accuracy for PM-GSA is not impacted by the number of retained components, since the estimation is performed directly for the original output. This is depicted in the figures with a *nrmse* curve corresponding to a straight line across the n_p values.

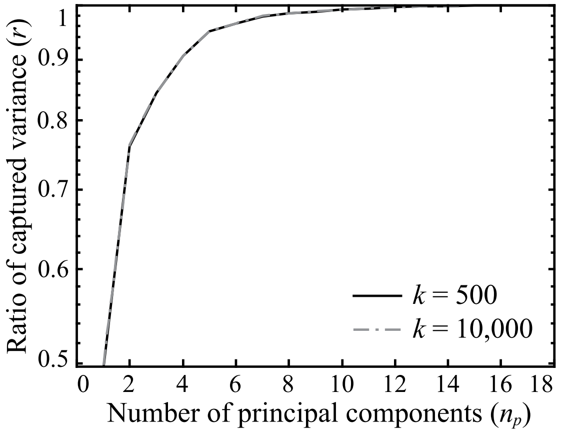

Figure 1. Portion of total variance of original output against the number of principal components retained, for implementation with k=500 and k=10,000 simulations (model evaluations) for the nine-story benchmark building example.

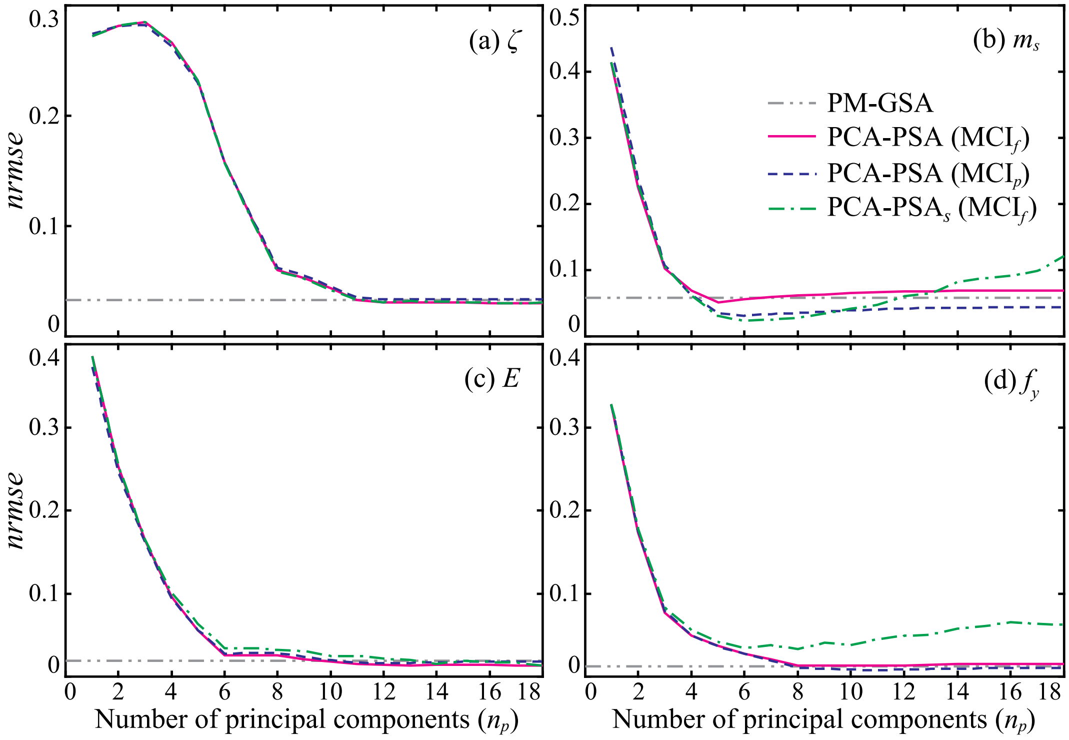

Figure 2. Error metric *nrmse* of the first-order Sobol' indices estimates for different GSA variants against the number of principal components retained, for implementation with k=500 simulations (model evaluations) for the nine-story benchmark building example.

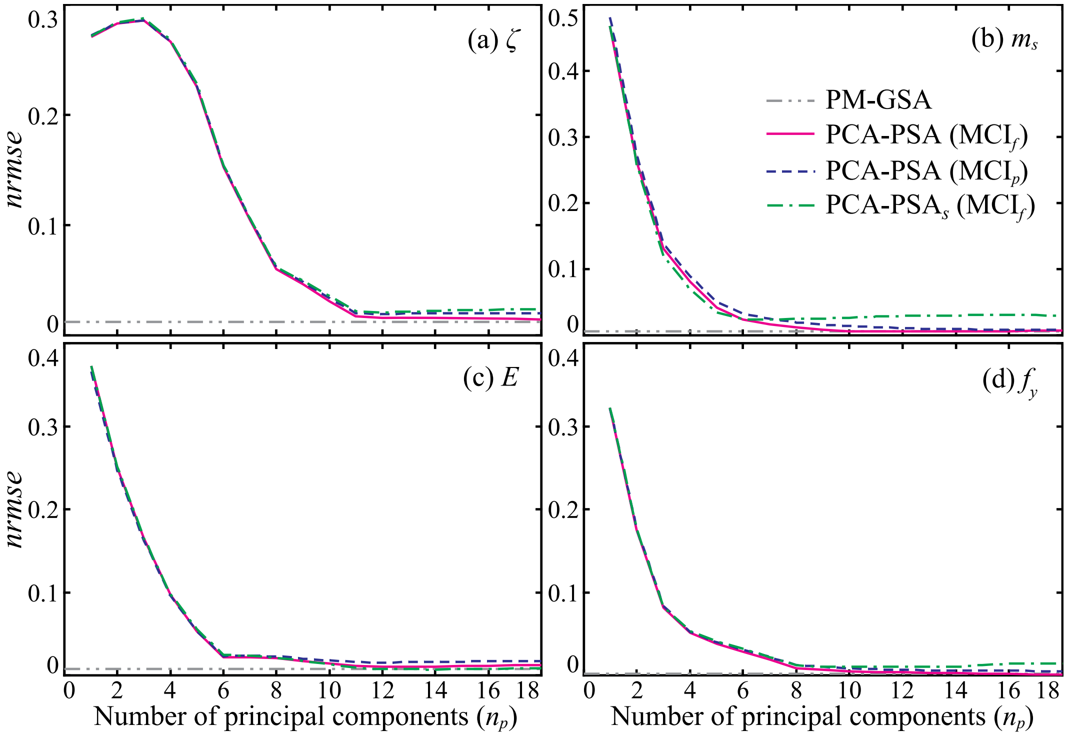

Figure 3. Error metric *nrmse* of the first-order Sobol' indices estimates for different GSA variants against the number of principal components retained, for implementation with k=10,000 simulations (model evaluations) for the nine-story benchmark building example.

Results in Figure 2 and Figure 3 clearly illustrate that, as expected, as the number of principal components increases the accuracy of the PCA-based GSA (PCA-PSA in these figures) improves, and that this accuracy saturates (no further improvement) by the time the variance of the retained components reaches a value of 99%. This trend agrees with results in [29], that similarly leveraged PCA to accommodate GSA for problems with large dimensional outputs. The divergence for PCA-PSA$_s$ implementation for larger n_p values for the smaller k value (Figure 2) will be discussed later when a comparison across the variant PCA-PSA cases is examined. Evaluating the total accuracy for PCA-PSA, comparisons to PM-GSA indicate that the PCA-based dimensionality reduction does not impact the GSA estimates: provided that a sufficient number of components is retained, the recommended implementations (this does not refer to PCA-PSA$_s$) perform equally well or better than PM-GSA. This demonstrates that any errors in the Sobol' indices estimation stem from the GMM-based approximation of the conditional statistics (the PM-GSA foundation), and not the dimensionality reduction or the modification of the output space that the approximation is developed on. Though the minimal impact by the PCA dimensionality reduction is expected when a sufficient number of latent components is used, the negligible influence by the output space the probability model approximation is developed on, is an important outcome, and it validates the proposed combination of PCA with the original PM-GSA framework: either if the probability model-based GSA is formulated for the original output (PM-GSA curve in the figures) or for the latent output (PCA-

PSA curves in the figures), the results in Figures 2 and 3 clearly demonstrate the same degree of accuracy. Note that for input ζ the initial reduction in accuracy as n_p increases should be attributed to a reduction of the quality of the GMM fit for the initially retained output components when examining the joint PDF for this specific input variable. As this trend pertains only to small n_p values, with large PCA truncation errors, and is quickly corrected, it should not be a concern for the overall PCA-PSA framework implementation.

Comparing now across the different PCA-PSA variants, we can observe that MCI$_p$ and MCI$_f$ implementations yield similar accuracy for PCA-PSA, while PCA-PSA$_s$ provides worse results and actually demonstrates a divergent behavior for the lower number of simulations case (Figure 2) as the number of principal components increases. The lower accuracy for PCA-PSA$_s$ should be attributed to the larger dimensional space considered for its implementation for the probability-model fit (n_c+n_p dimensional PDF), and the fact that as the number of principal components increases, that dimension is also increasing. For the PCA-PSA implementations the latter does not hold, with the probability-model fit established for n_c+1 dimensional PDFs for MCI$_f$ and n_c+1 or n_c+2 dimensional PDFs for MCI$_p$. Note that for the first-order indices n_c is equal to 1, which means that the addition of n_p dimensions to the probability model (for PCA-PSA$_s$) instead of 1 or 2 (for PCA-PSA) has a significant effect. The difference in the dimension of the probability-model fit between PCA-PSA and PCA-PSA$_s$ is the reason why there is a larger accuracy improvement by the increase of the number of simulations k (compare Figure 3 results to Figure 2 results) for PCA-PSA$_s$ compared to PCA-PSA, and the divergent behavior for PCA-PSA$_s$ for larger n_p values does no longer exist for the larger k: for PCA-PSA even with the lower number of simulations, the information available is sufficient for the GMM fit, whereas for PCA-PSA$_s$ the higher dimensionality of the joint-PDF means that larger amount of information is needed to accommodate a good fit (curse of dimensionality), creating a more substantial relative influence when the value of k increases.

Figure 4 presents results for the first-order indices for all 18 individual outputs for the case with k=500 model evaluations, for all different GSA variants, including the reference double-loop MCI implementation. The results for all PCA-based implementations correspond to the number of principal components for r_o=99.9% threshold. Trends are consistent with the ones reported in Figure 2 and Figure 3 and do not exhibit any differences across the individual outputs. Across all outputs, both PCA-PSA variants offer estimates close to the reference results and similar to the implementation without PCA (PM-GSA). Comparison across the two PCA-PSA variants, shows that the estimates from the MCI$_p$ implementation are closer to the PM-GSA predictions. This is not surprising, since as discussed earlier the original PM-GSA adopts a Monte Carlo integration approach that is identical to the MCI$_p$.

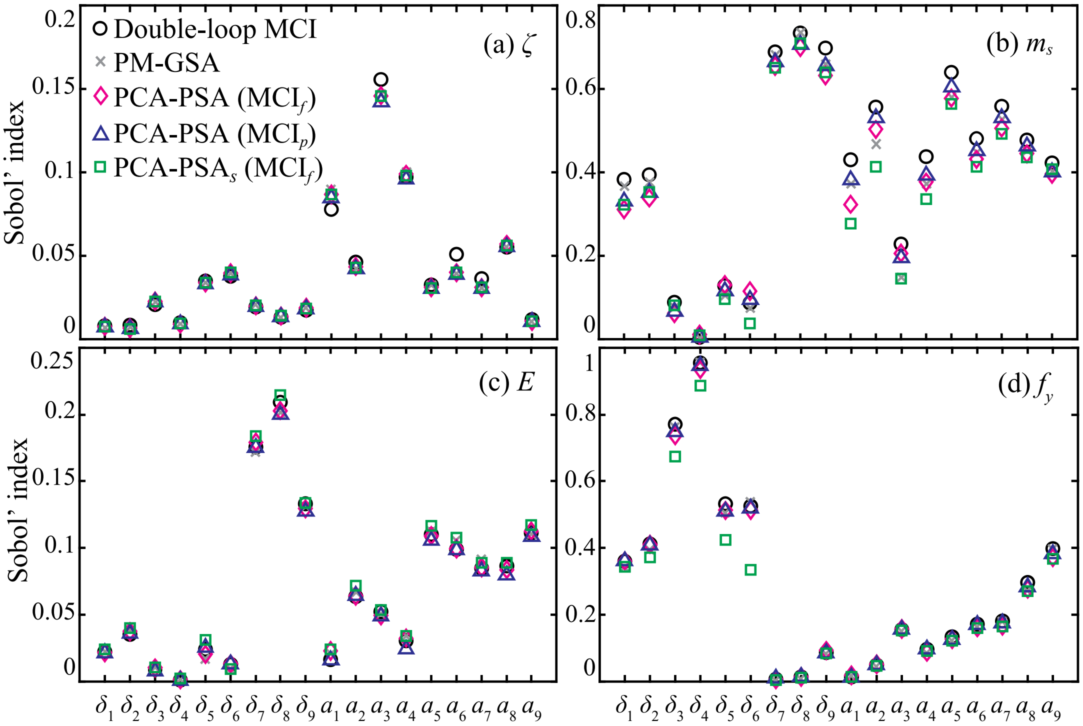

Figure 4. Comparison of individual first-order Sobol' indices between different GSA variants, for implementation with k=500 simulations (model evaluations) for the nine-story benchmark building example. Number of retained components corresponds to r_o=99.9% for the PCA-based implementations.

Table 1. Error metric *nrmse* of different Sobol' indices across different GSA variants, for implementations with different number of simulations k (model evaluations) for the nine-story benchmark building example. Number of retained components corresponds to r_o=99.9% for the PCA-based implementations.

nrmse	k	First-order indices				Higher-order indices		Total-effect indices			
		ζ	m_s	E	f_y	m_s, E	m_s, E, f_y	ζ	m_s	E	f_y
PM-GSA	500	0.033	0.057	0.019	0.013	0.211	0.497	0.739	0.026	0.205	0.121
	1000	0.020	0.051	0.015	0.013	0.098	0.349	0.508	0.020	0.144	0.074
	10,000	0.012	0.006	0.008	0.002	0.054	0.266	0.165	0.004	0.044	0.023
PCA-PSA (MCI$_f$)	500	0.030	0.069	0.014	0.015	0.224	0.622	0.650	0.035	0.173	0.135
	1000	0.021	0.044	0.016	0.015	0.169	0.462	0.486	0.025	0.138	0.100
	10,000	0.015	0.006	0.012	0.002	0.061	0.228	0.169	0.004	0.052	0.024
PCA-PSA (MCI$_p$)	500	0.033	0.043	0.018	0.010	0.233	0.405	0.676	0.020	0.139	0.108
	1000	0.028	0.043	0.024	0.012	0.146	0.271	0.547	0.023	0.096	0.072
	10,000	0.020	0.008	0.017	0.005	0.071	0.194	0.187	0.007	0.036	0.022
PCA-PSA$_s$ (MCI$_f$)	500	0.030	0.098	0.025	0.064	0.307	0.442	1.388	0.069	0.275	0.123
	1000	0.022	0.058	0.013	0.026	0.247	0.289	0.896	0.057	0.157	0.121
	10,000	0.024	0.031	0.008	0.015	0.070	0.332	0.346	0.013	0.076	0.049

Table 1 further summarizes the *nrmse* values for first-order Sobol' indices but also presents values for some higher-order (the ones that were identified to have higher values) and for all total-effect Sobol' indices, for all considered values of model evaluations *k*. Similar to Figure 4 all different GSA variants are presented, with the results reported for the PCA-based implementations corresponding to the number of principal components for r_o=99.9%. Results for higher-order and total-effect indices show reduced accuracy. This should be attributed to the effect of the probability model-based approximation, and not to the PCA dimensionality reduction, as evident by the fact that the same trends hold for PM-GSA as for PCA-PSA. This is related fundamentally to the dimensionality of the fitted distribution for the GMM; this dimensionality is dependent on the n_c value, which is proportional to the order of the index considered. Note that for the higher-order indices greater reduction of accuracy is observed compared to the total-effect indices that similarly involve larger dimensional probability model fits. This greater should be additionally attributed to the smaller, in general, values these indices take, as they quantify correlation effects that are in general smaller than the total-effect or the first-order values of individual inputs. All these trends agree with the ones reported in [25], and relate to the greater challenges associated with fitting probability models in higher dimensions. Such trends will be similar for any sensitivity indices that involve large dimensional inputs, including, as discussed earlier, first-order indices for groups of inputs. Focusing on the effect of the PCA dimensionality reduction, the same trends that hold for the first-order indices with respect to the MCI$_f$ and MCI$_p$ implementations for PCA-PSA or the PCA-PSA$_s$ performance, hold for the higher-order and total-effect indices.

Finally, as the number of simulations *k* increases, the accuracy of the GSA estimates increases. This pattern holds across all indices, but the impact is greater for the higher-order and the total-effect indices. The higher dimensionality of the probability model fit for those indices is what contributes to this greater impact. Note that an identical influence was identified for PCA-PSA$_s$ earlier, for the exactly same reasons.

4.2 Forecasting of peak surge during Superstorm Sandy

The second example corresponds to a high-dimensional output case and examines the real-time probabilistic forecasting of storm surge. A detailed description of the problem formulation is available in [47]. The model input **x**, in this case, corresponds to the cross and along track variability, Δs_{cross} and Δs_{along} respectively, the intensity, Δv_w, and the size, ΔR_{mw}, of a landfalling storm, and the objective is to provide probabilistic predictions for the expected storm surge a few days before the storm makes landfall [48]. The probability distribution for **x** is based on historical forecast errors and corresponds to, or can be converted to, Gaussian distributions. The output **y** corresponds to the peak surge elevation (storm surge over the node elevation), across the entire spatial grid encompassed by the underlying numerical model (used to predict the storm surge). The specific storm examined here corresponds to Superstorm Sandy and the forecast examined is for the National Weather Service Advisory 20, which roughly corresponds to 72 hours before landfall. The nominal storm track will be shown in some of the figures presented later, whereas an illustration of the uncertainties associated with this advisory can be found in [47]. The numerical model utilized for the surge predictions corresponds to a surrogate model approximation for the north Atlantic coast [49] informed by simulations [50] of ADCIRC (Advanced Circulation Model for Shelves, Coastal Seas, and Estuaries) [51]. For the implementation discussed here, this approximation is considered to correspond to the exact numerical model, and within PCA-PSA is only utilized to perform the k simulations to obtain the sample set [**X**, **Y**]. Its use instead of ADCIRC is necessitated by the need to estimate reference solutions for the Sobol' indices, something that requires a large number of simulations. The part of the ADCIRC grid within the domain of impact of the storm includes n_y=1,374,934 nodes (dimension of output). These correspond to nodes with at least 5% probability of being inundated for this specific advisory. Three different values for the number of total simulations will be examined, k=500, k=1000 and k=10,000, for the proposed data-driven PCA implementation. The first two should be considered reasonable values for real-time surge forecasting applications [47], while the third, larger value is examined to better investigate the impact on accuracy when larger amount of data is available.

For this example, the output also poses some unique challenges as it is constrained to be greater than zero, with zero indicating the node being dry (not inundated since surge is equal to the node elevation). A significant portion of the domain (number of outputs) remains dry for multiple storm simulations (many of the inputs $\mathbf{x}^s$). This bounded behavior for **y** poses challenges for the GMM fit since the latter establishes an unbounded PDF fit to the provided data. These challenges can substantially reduce the accuracy of the respective conditional statistics estimates. The degree of accuracy reduction will ultimately depend on the magnitude of the boundary effects and will differ across the different outputs (nodes in the domain). For this reason, the implementation of PM-GSA is considered only for the outputs that remain inundated across all storm simulations. Even for some outputs belonging to this group, the saturation of surge output values close to zero might still impose some challenges, as will be also discussed later. Note that one could have tried different approaches to accommodate the complex output behavior, and these challenges should not be regarded as deficiencies of the PM-GSA formulation. Here the direct implementation of the original PM-GSA framework is considered [25], instead of investigating appropriate modifications. For the PCA-PSA implementation, these challenges are remedied by the transformation to the latent output space, since bounded behavior for **z** is not necessarily expected. As such, there is no reason to anticipate substantial adverse effects of the GMM fit for the latent outputs, originating from the bounded behavior of the original output. For this reason, the application of PCA-PSA is considered for the entire original output. To accommodate comparisons to PM-GSA, results will be reported for two different groups of output: the entire output, denoted as *entire domain* herein, and a portion of the nodes that are inundated (output not saturating at zero) across all simulations, denoted as *inundated domain* herein. As discussed earlier, PM-

GSA is considered only for the latter group. The definition of *inundated domain* is based on k=500 simulations, which ultimately means that the probability of these nodes being dry is less than 1/500=0.2%. Also for accommodating the computational challenges associated with PM-GSA for application across all outputs (nodes) for the large value of simulations (k=10,000), for the *inundated domain* implementation, a simplification is established, considering only n_y=100,000 randomly chosen outputs that are inundated across all simulations, instead all outputs belonging in that category (inundated across all simulations), corresponding in this case to 800,000 nodes. Since choice is random and is a big proportion of the available nodes, this simplification in the implementation should not be considered to impact results. Note that the PCA-PSA formulation was performed only once, for the *entire domain*. Simply when accuracy results are presented for the *inundated domain*, the predictions for the respective outputs are only used in the *nrmse* definition according to Eq. (22). Accuracy could potentially be improved if PCA-PSA was considered strictly for the *inundated domain*.

Figure 5 presents, similarly to Figure 1, the ratio of captured variance as a function of the number of retained principal components for implementations with k=1000 or 10,000 simulations (model evaluations) for the entire domain. Based on the results of the figure, for r_o=99%, n_p=11 number of components are needed whereas for r_o=99.9% n_p=40 number of components are needed for both values of k =1000 or 10,000. Despite the larger dimension of the output, we observe that there is no substantial influence of the number of simulations k on the variation of r with respect to the number of principal components.

Figures 6 and 7 show the accuracy improvement for the estimation of the first-order Sobol' indices, quantified through *nrmse* value, as the number of retained principal components (n_p) increases for the case with k=1000 and k=10,000 simulations, respectively (similar to Figure 2 and Figure 3 presentations) for the entire domain. PM-GSA predictions are not included in these figures. Figure 8 presents similar results for k=1000 for the *inundated domain*. PM-GSA results are included in this figure, while results for PCA-PSA$_s$ are omitted due to the poor performance that is exhibited for this implementation when considering the application to the *entire domain* (in Figure 6 and Figure 7). Table 2 summarizes the *nrmse* values for first-order Sobol' indices but also presents values for some higher-order (the ones that were identified to have higher values) and for all total-effect Sobol' indices, for all considered values of model evaluations k for the *entire domain*. As in Figure 6 and Figure 7, no PM-GSA results are reported in this table. Table 3 presents similar results for the *inundated domain*. The same variant types considered in Figure 8 are included in this table. Finally, Figure 9 shows the spatial variability of all first-order Sobol' indices within the geographic domain of impact of Superstorm Sandy (focusing on New Jersey and New York) and presents comparisons between the reference results and the results established by the PCA-PSA implementations. Figure 10 shows identical results for the total-effect Sobol' indices. In both Figure 9 and Figure 10, the storm track forecast for Advisory 20 is also shown with a red line. For the PCA-based implementations, the results in Tables 2 and 3, and Figures 9 and 10 correspond to the number of principal components for r_o=99.9%.

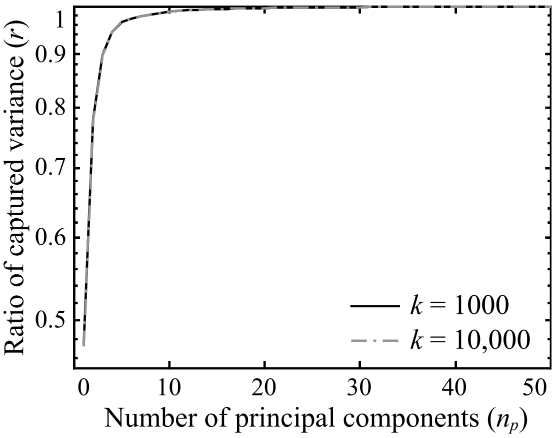

Figure 5. Portion of total variance of original output against the number of principal components retained, for implementation with k=1000 and k=10,000 simulations (model evaluations) for the storm-surge forecasting example.

Results verify the general trends, reported in the previous example, in this case in a higher-dimensional implementation (over million of outputs), but also present some differences, especially with respect to the comparisons with PM-GSA. Like in the previous example, as the number of principal components increases the accuracy of the PCA-based GSA (PCA-PSA in these figures) improves, and this accuracy saturates by the time the variance of the retained components reaches a value of 99%, verifying that the PCA-based dimensionality reduction does not impact the probability-model GSA implementation as long as a sufficient number of components is retained. Comparison to the PM-GSA results in Figure 8 and Table 2 indicates that PCA-PSA typically outperforms PM-GSA for this example. This should be attributed, as discussed earlier, to the challenges encountered by the GMM fit of PM-GSA due to the bounded behavior for the original surge output **y**. Since such bounded behavior is not necessarily expected for the latent output **z**, PCA-PSA ends up slightly outperforming PM-GSA. This discussion indicates that in this case the transformed (latent) output space the probability model approximation is developed on has an overall positive impact on the overall accuracy, further validating the proposed integration of PCA within the original PM-GSA formulation. This positive impact is also clearly evident when comparing the accuracy results between the *entire domain* (Figure 6 for k=1000 or Table 2) and the *inundated domain* (Figure 8 for k=1000 or Table 3). Similar level of accuracy is accomplished across these sets of outputs, indicating the latent output transformation clearly accommodates complexities associated with nodes that have remained dry for many of the storm simulations.

In general, the comparisons indicate that any errors in the Sobol' indices estimates should be attributed to challenges associated with the probability-based approximation for estimation of the conditional variance and covariance statistics, and not with the addition of the PCA step. These challenges are greater for higher-order and total-effect indices, as clearly shown in Tables 2 and 3, owing to the larger dimensionality of the fitted distribution for the GMM; as shown earlier this dimensionality is related to the n_c value, which is proportional to the order of the index considered. Examining further the accuracy variation with the increase of the number of PCA components in Figures 6-8, we again observe some initial reduction in accuracy for some indices as n_p initially increases. Like in the previous example, this should be attributed to a reduction of the quality of the GMM fit for the originally retained output components when examining the joint PDF, and is not of serious concern since it is associated with n_p values for which a large PCA truncation errors exist (cases that should be avoided anyway). Also, the divergence of PCA-PSA$_s$ estimates is more evident in this example. This was expected since this divergence is related, as identified earlier, to challenges associated with the large dimensionality of the fitted PDF across all retained components for PCA-PSA$_s$ (n_c+n_p dimensional PDF). Since the n_p values that need to be considered increase in this example, owing to the larger dimensionality of the output n_y, the associated challenges are magnified.

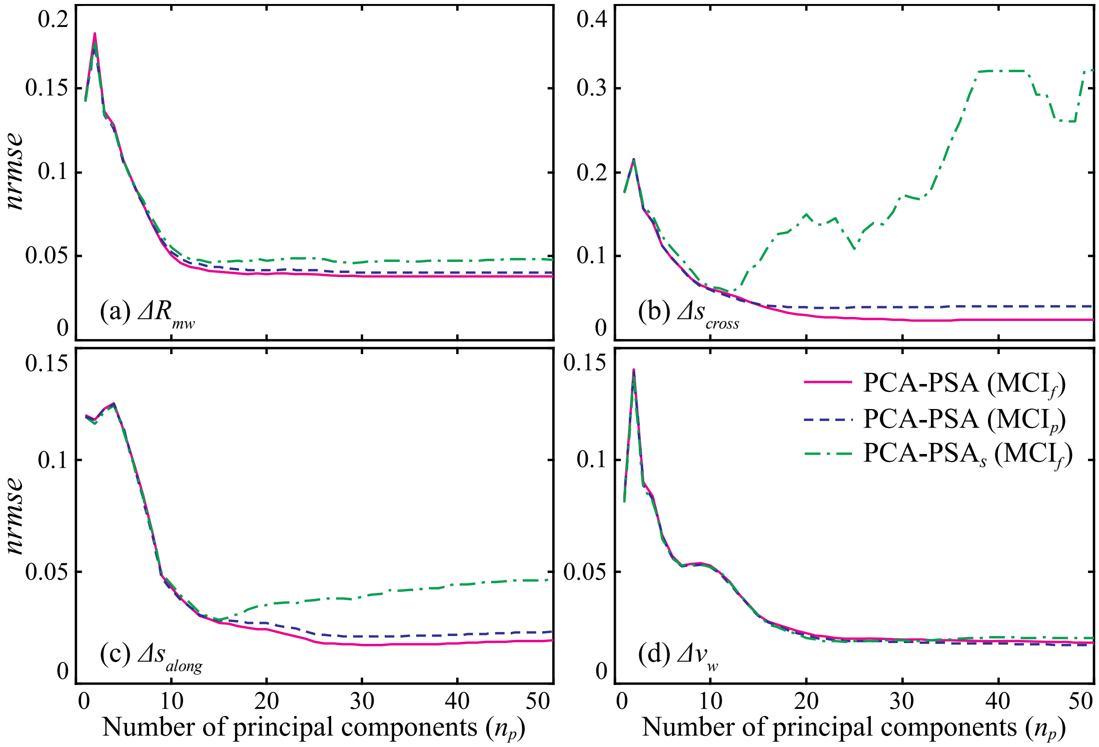

Figure 6. Error metric *nrmse* of the first-order Sobol' indices estimates for different GSA variants against the number of principal components retained, for implementation with k=1000 simulations (model evaluations) for the storm-surge forecasting example for the *entire domain*.

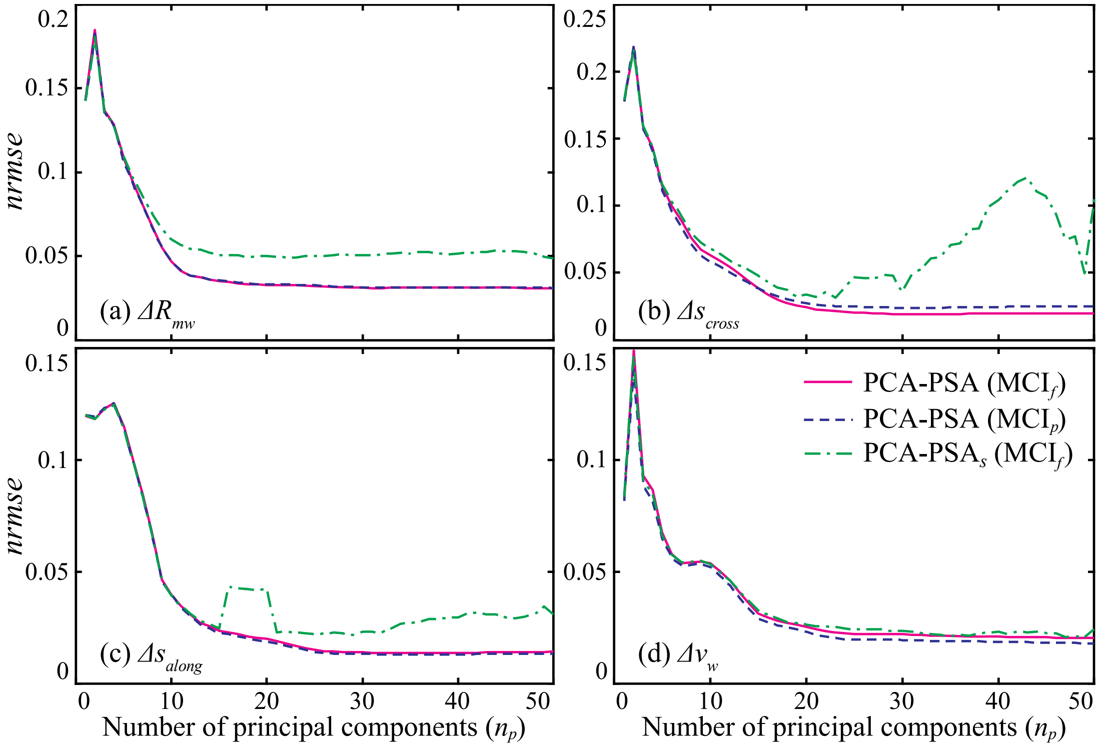

Figure 7. Error metric *nrmse* of the first-order Sobol' indices estimates for different GSA variants against the number of principal components retained, for implementation with k=10,000 simulations (model evaluations) for the storm-surge forecasting example for the *entire domain*.

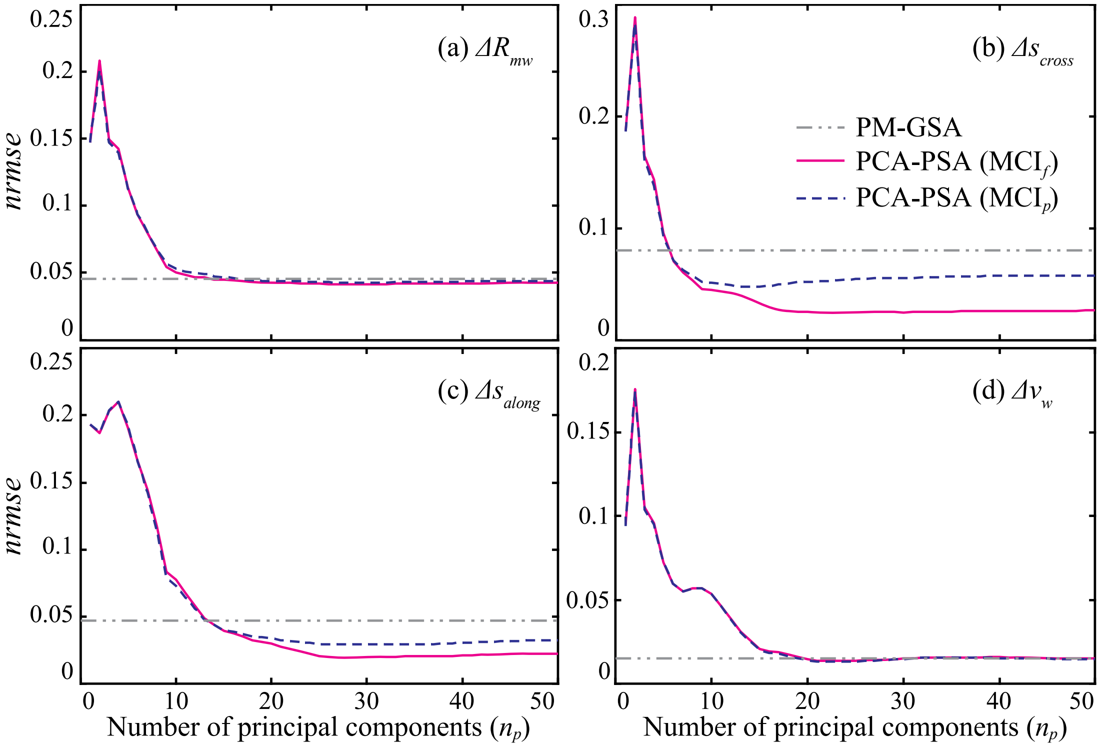

Figure 8. Error metric *nrmse* of the first-order Sobol' indices estimates for different GSA variants against the number of principal components retained, for implementation with k=1000 simulations (model evaluations) for the storm-surge forecasting example for the *inundated domain*.

Table 2. Error metric *nrmse* of different Sobol' indices across different GSA variants, for implementations with different number of simulaitons *k* (model evaluations) for the storm-surge forecasting example for the *entire domain*. Number of retained components corresponds to r_o=99.9% for the PCA-based implementations.

nrmse	*k*	First-order indices				Higher-order indices		Total-effect indices			
		ΔR_{mw}	Δs_{cross}	Δs_{along}	Δv_w	Δs_{cross}, Δs_{along}	ΔR_{mw}, Δs_{cross}, Δv_w	ΔR_{mw}	Δs_{cross}	Δs_{along}	Δv_w
PCA-PSA (MCI$_f$)	500	0.040	0.026	0.036	0.020	0.104	0.119	0.128	0.059	0.133	0.136
	1000	0.038	0.024	0.018	0.019	0.104	0.122	0.084	0.050	0.107	0.106
	10,000	0.031	0.020	0.014	0.021	0.086	0.118	0.038	0.019	0.068	0.069
PCA-PSA (MCI$_p$)	500	0.046	0.084	0.033	0.020	0.105	0.117	0.146	0.062	0.138	0.158
	1000	0.040	0.040	0.022	0.018	0.108	0.126	0.087	0.054	0.086	0.109
	10,000	0.031	0.025	0.013	0.019	0.077	0.109	0.049	0.034	0.060	0.050
PCA-PSA$_s$ (MCI$_f$)	500	0.048	0.318	0.045	0.022	0.112	0.108	0.475	0.098	0.455	0.454
	1000	0.048	0.321	0.044	0.021	0.273	0.114	0.474	0.103	0.460	0.241
	10,000	0.051	0.080	0.031	0.025	0.109	0.158	0.153	0.048	0.099	0.172

Table 3. Error metric *nrmse* of different Sobol' indices across different GSA variants, for implementations with different number of simulations *k* (model evaluations) for the storm-surge forecasting example for the *inundated domain*. Number of retained components corresponds to r_o=99.9% for the PCA-based implementations.

nrmse	*k*	First-order indices				Total-effect indices			
		ΔR_{mw}	Δs_{cross}	Δs_{along}	Δv_w	ΔR_{mw}	Δs_{cross}	Δs_{along}	Δv_w
PM-GSA	500	0.051	0.127	0.056	0.018	0.181	0.062	0.292	0.310
	1000	0.046	0.081	0.047	0.015	0.112	0.053	0.196	0.209
	10,000	0.040	0.058	0.019	0.015	0.098	0.030	0.126	0.154
PCA-PSA (MCI$_f$)	500	0.043	0.027	0.054	0.018	0.117	0.049	0.145	0.203
	1000	0.042	0.026	0.021	0.016	0.084	0.048	0.126	0.155
	10,000	0.036	0.020	0.019	0.016	0.047	0.019	0.109	0.122
PCA-PSA (MCI$_p$)	500	0.050	0.119	0.050	0.019	0.187	0.057	0.201	0.281
	1000	0.044	0.058	0.031	0.015	0.108	0.055	0.113	0.174
	10,000	0.035	0.032	0.015	0.015	0.057	0.037	0.080	0.081

Comparing now the two PCA-PSA variants, we observe bigger differences between the MCI$_p$ and MCI$_f$ implementations compared to the previous example, especially for lower values of *k*. Though for larger values of *k* results for MCI$_p$ and MCI$_f$ are quite similar for all examined sensitivity indices, as evident especially in the comparisons in Tables 2 and 3, for lower values of *k*, MCI$_f$ clearly outperforms MCI$_p$. Evidently, the relative greater challenges expected for the GMM fit for this example impact the Monte

Carlo integration when the MCI$_p$ formulation is adopted, leading to the relative reduction in accuracy for the smaller *k* values, when that GMM fit is expected to be of lower quality. Examining additional trends in Tables 2 and 3 for the influence of the number of simulations, as expected as *k* increases, the accuracy of the GSA estimates increases. This pattern holds across all indices, but the impact is greater for the higher-order and the total-effect indices, and for the PCA-PSA$_s$ and PM-GSA implementations. The influence of *k* on higher-order and total-effect indices and on the PCA-PSA$_s$ implementation is attributed to the fact that all these instances require higher dimensional fitted GMMs, which are impacted by the n_c value for the higher-order and total-effect indices and also by the n_p value for the PCA-PSA$_s$ implementation. This higher dimensionality provides relatively larger benefits when the value of *k* increases. For first-order indices evaluated through the PCA-PSA implementation, even lower number of simulations provide sufficient information for the GMM fit, contributing to a relatively smaller impact by the *k* value. The influence of *k* on the PM-GSA implementation is attributed to the challenges the GMM fit encounters with the bounded surge response for some of the output components. Larger values of *k* evidently help alleviate some of the associated challenges, contributing to the improvement in the sensitivity index approximation even for first-order indices. This is a similar trend to the one identified earlier for the MCI$_p$ formulation for PCA-PSA, though the impact for PM-GSA is more substantial, as the challenges for the GMM fit in the original output (as opposed to latent output) are expected to be substantially larger.

Overall the comparisons across both examples reveal that despite its computational advantages, requiring only one GMM fit, PCA-PSA$_s$ is not a viable alternative due to the reduction in accuracy even when the number of simulations is high. For PCA-PSA, MCI$_f$ implementation should be preferred due to its demonstrated better robustness for lower *k* (model simulations) and significant computational benefits, requiring only n_p GMM fits instead of $n_p(n_p+1)/2$, which for larger number of retained components drastically reduces the computational burden of the PCA-PSA implementation.

Finally, results in Figure 9 and Figure 10 demonstrate the insights that can be established through GSA in this application, showing an important spatial variation of the sensitivity indices across the geographic domain. Depending on the specific part of the domain, different types of forecast errors (different components of input vector **x**) are identified as more important in influencing the storm surge (Figure 9), whereas for big parts of the domain some of them (Δs_{along}) are identified as entirely unimportant based on the total-effect indices (Figure 10). The lower accuracy of the PCA-PSA$_s$ implementation is also very evident in these figures, providing in some instances (check the larger values of total-effect indices for Δs_{along}, for example) erroneous sensitivity trends. On the other hand, all PCA-PSA variants offer very good matches to the reference results, and identify the correct spatial sensitivity trends. This is true for both first-order and total-effect sensitivity indices, indicating that the larger estimation error for the latter type of indices does not alter the insights that will be obtained from the GSA implementation for understanding the relative importance of the different types of forecast errors as well as the spatial distribution of this importance. This offers another validation of the proposed PCA-PSA formulation, demonstrating that any estimation errors have a small influence on the underlying objectives of the GSA implementation.

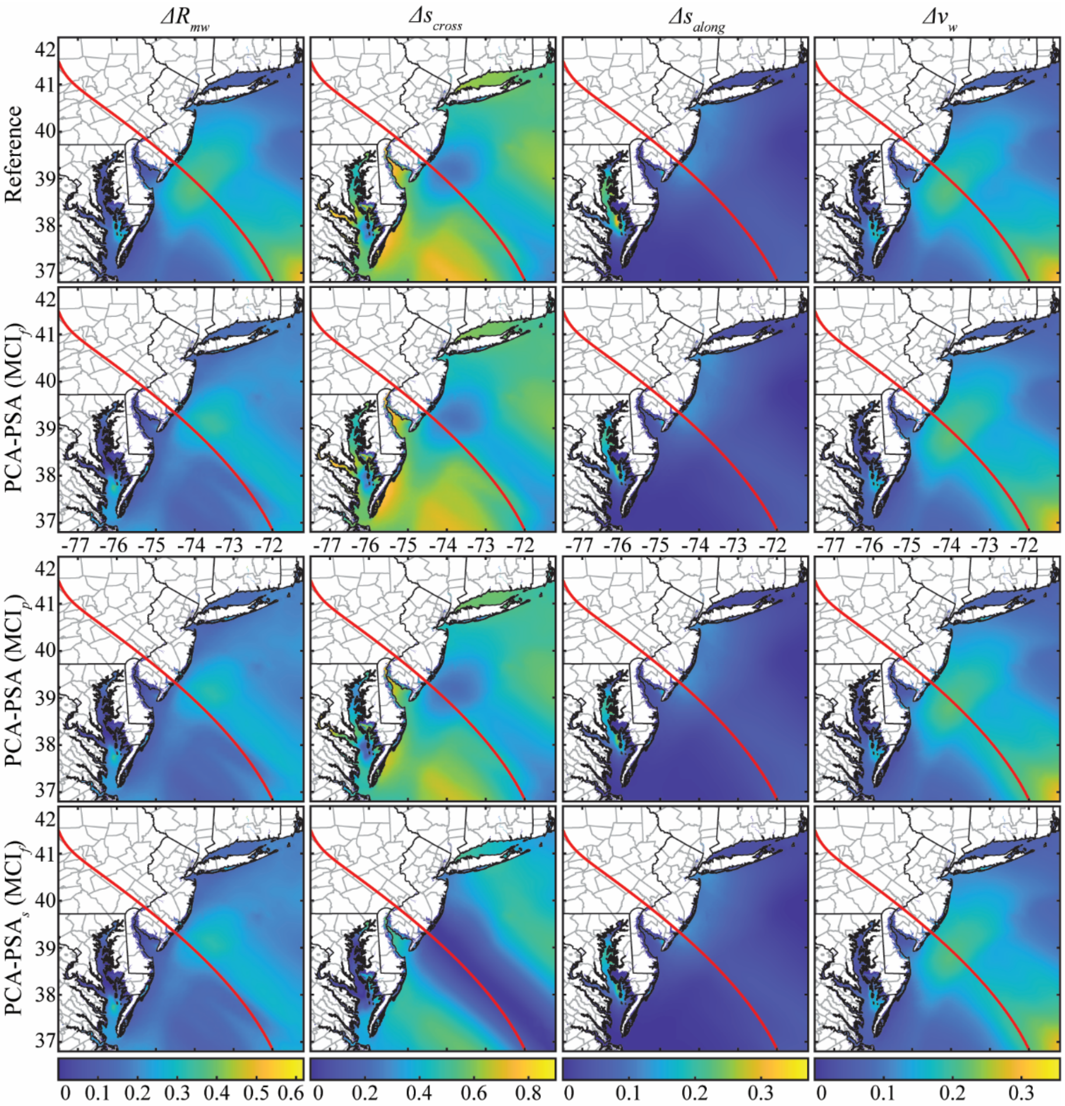

Figure 9. Spatial variability of first-order Sobol' indices for different GSA variants, for implementation with *k*=1000 simulations (model evaluations) for the storm-surge forecasting example. Number of retained components corresponds to r_o=99.9% for the PCA-based implementations. The red line corresponds to the forecasted storm track for Advisory 20.

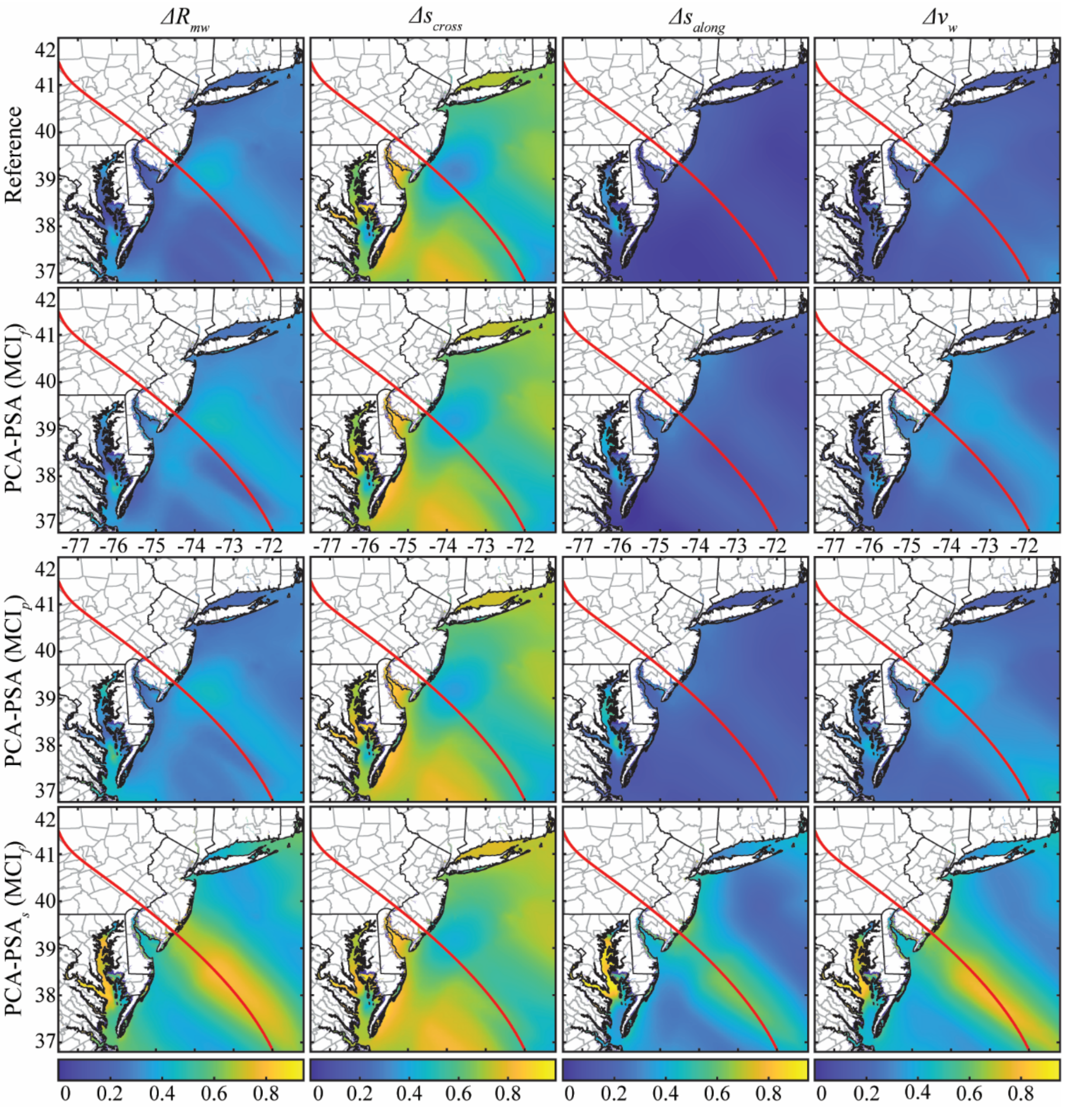

Figure 10. Spatial variability of total-effect Sobol' indices for different GSA variants, for implementation with k=1000 simulations (model evaluations) for the storm-surge forecasting example. Number of retained components corresponds to r_o=99.9% for the PCA-based implementations. The red line corresponds to the forecasted storm track for Advisory 20.

5. Conclusions

This paper examined the efficient variance-based global sensitivity analysis (GSA), quantified through the estimation of first-/higher-order and total-effect Sobol' indices, for applications involving complex numerical models and high-dimensional outputs. An efficient data-driven framework, termed PCA and Probability model-based sensitivity analysis (PCA-PSA), was established by using as foundation the dimensionality reduction approach in [29] and by proposing an alternative for the surrogate model–based formulation of that study, adopting and extending the probability model-based GSA of [25]. Following [29], principal component analysis (PCA) is first implemented, restricting the computational implementation to the estimation of statistics (variance and covariance) for the latent outputs (principal components), instead of the original high-dimensional output. The probability model-based GSA [25] is then extended to estimate these statistics, relying on the development of a Gaussian mixture model (GMM) to approximate the joint probability density function between some subset of the input vector (dependent upon the Sobol' index estimated), and each latent output, or each pair of latent outputs. Different Monte Carlo integration schemes were examined to accommodate the probability model-based estimation of the different statistics: MCI$_f$ that requires development of a total of n_p GMMs, where n_p denotes the number of retained components, and MCI$_p$ that requires the development of a total of $n_p(n_p+1)/2$ GMMs. The computationally intensive part of the PCA-PSA algorithm is the GMM fit to the data. For applications with large dimensional outputs, for which substantial discrepancy is expected between the number of individual outputs and the number of retained components, this approach can offer substantial computational savings.

The efficiency and accuracy of the proposed algorithm were demonstrated in two examples, the first considering the sensitivity of different peak engineering demand parameters for a nine-story benchmark building with uncertain model properties, exposed to earthquake acceleration at its base, and the second examining the sensitivity of the estimated peak-surge to the storm forecast variability during Superstorm Sandy. Results showed that the PCA-based dimensionality reduction does not impact the probability-model GSA implementation as long as a sufficient number of components is retained. The output space (latent space versus original output space) the probability model approximation is developed on does not have an impact on the overall accuracy, validating the proposed integration of PCA within the original probability model-based GSA formulation. Errors in the Sobol' indices estimates were shown to be connected to challenges associated with the GMM-fit to the available data, not related to the dimensionality reduction or output transformation established through PCA. Additionally, for applications for which the original output behavior creates challenges for the GMM fit, transformation to the latent output space can actually be beneficial. This was clearly demonstrated in addressing dry nodes in the Superstorm Sandy application. An alternative formulation of the PCA-PSA approach was also investigated, requiring only one GMM fit that can be used across all estimated statistics, but was shown to provide a significant reduction in accuracy and so should be avoided. Finally, both MCI$_f$ and MCI$_p$ implementations were shown to provide in general similar accuracy, with MCI$_f$ even outperforming MCI$_p$ for cases where the GMM fit encounters challenges. When combined with the higher computational efficiency accommodated through MCI$_f$, trends indicate a strong preference for it.

Acknowledgments

This research was funded by the National Oceanographic and Atmospheric Administration (NOAA), grant number NA19OAR0220089. The views and opinions expressed in this paper are those of the authors and do not represent NOAA. Authors would like to thank the Army Corp of Engineers, Coastal Hydraulics Laboratory of the Engineering Research and Development Center for providing access to the databases that were used to develop the surrogate models for the storm surge predictions.

Data Availability

Data utilized for the creation of the surrogate models for the second example were provided by the Army Corps of Engineers through the Coastal Hazards System https://chs.erdc.dren.mil/default.aspx. Data and models utilized for the first example are available through the corresponding author upon reasonable request.

References

[1] Fu G, Kapelan Z, Reed P. Reducing the complexity of multiobjective water distribution system optimization through global sensitivity analysis. Journal of Water Resources Planning and Management. 2012;138:196-207.

[2] Vetter C, Taflanidis A. Comparison of alternative stochastic ground motion models for seismic risk characterization. Soil Dynamics and Earthquake Engineering. 2014;58:48-65.

[3] Hu Z, Mahadevan S. Global sensitivity analysis-enhanced surrogate (GSAS) modeling for reliability analysis. Structural and Multidisciplinary Optimization. 2016;53:501-21.

[4] Saltelli A. Sensitivity analysis for importance assessment. Risk Anal. 2002;22:579-90.

[5] Papaioannou I, Straub D. Variance-based reliability sensitivity analysis and the FORM α-factors. Reliability Engineering & System Safety. 2021;210:107496.

[6] Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D et al. Global sensitivity analysis: the primer: John Wiley & Sons; 2008.

[7] McRae GJ, Tilden JW, Seinfeld JH. Global sensitivity analysis—a computational implementation of the Fourier amplitude sensitivity test (FAST). Comput Chem Eng. 1982;6:15-25.

[8] Xu C, Gertner G. Extending a global sensitivity analysis technique to models with correlated parameters. Computational Statistics & Data Analysis. 2007;51:5579-90.

[9] Lewandowski D, Cooke RM, Tebbens RJD. Sample-based estimation of correlation ratio with polynomial approximation. ACM Transactions on Modeling and Computer Simulation (TOMACS). 2007;18:1-17.

[10] Jia G, Taflanidis AA. Sample-based evaluation of global probabilistic sensitivity measures CoStr. 2014;144:103-18.

[11] Zhu X, Sudret B. Global sensitivity analysis for stochastic simulators based on generalized lambda surrogate models. Reliability Engineering & System Safety. 2021;214:107815.

[12] Sobol' IM. On sensitivity estimation for nonlinear mathematical models. Matematicheskoe Modelirovanie. 1990;2:112-8.

[13] Iooss B, Lemaître P. A review on global sensitivity analysis methods. Uncertainty management in simulation-optimization of complex systems: Springer; 2015. p. 101-22.

[14] Sobol' IM. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Mathematics and computers in simulation. 2001;55:271-80.

[15] Homma T, Saltelli A. Importance measures in global sensitivity analysis of nonlinear models. Reliability Engineering & System Safety. 1996;52:1-17.

[16] Ökten G, Liu Y. Randomized quasi-Monte Carlo methods in global sensitivity analysis. Reliability Engineering & System Safety. 2021;210:107520.

[17] Tarantola S, Gatelli D, Mara TA. Random balance designs for the estimation of first order global sensitivity indices. Reliability Engineering & System Safety. 2006;91:717-27.

[18] Jia G, Taflanidis AA. Efficient evaluation of Sobol' sensitivity indices utilizing samples from an auxiliary probability density function. Engineering Mechanics. 2016;142:04016012.

[19] Li C, Mahadevan S. An efficient modularized sample-based method to estimate the first-order Sobolʹ index. Reliability Engineering & System Safety. 2016;153:110-21.

[20] Sudret B. Global sensitivity analysis using polynomial chaos expansions. Reliability engineering & system safety. 2008;93:964-79.

[21] Chen W, Jin R, Sudjianto A. Analytical variance-based global sensitivity analysis in simulation-based design under uncertainty. Journal of Mechanical Design 2005;127:875-86.

[22] Rohmer J, Lecacheux S, Pedreros R, Quetelard H, Bonnardot F, Idier D. Dynamic parameter sensitivity in numerical modelling of cyclone-induced waves: a multi-look approach using advanced meta-modelling techniques. Natural Hazards. 2016;84:1765-92.

[23] Kapusuzoglu B, Mahadevan S. Information fusion and machine learning for sensitivity analysis using physics knowledge and experimental data. Reliability Engineering & System Safety. 2021;214:107712.

[24] Antoniadis A, Lambert-Lacroix S, Poggi J-M. Random forests for global sensitivity analysis: A selective review. Reliability Engineering & System Safety. 2021;206:107312.

[25] Hu Z, Mahadevan S. Probability models for data-driven global sensitivity analysis. Reliability Engineering & System Safety. 2019;187:40-57.

[26] Becker W. Metafunctions for benchmarking in sensitivity analysis. Reliability Engineering & System Safety. 2020;204:107189.

[27] Campbell K, McKay MD, Williams BJ. Sensitivity analysis when model outputs are functions. Reliability Engineering & System Safety. 2006;91:1468-72.

[28] Lamboni M, Monod H, Makowski D. Multivariate sensitivity analysis to measure global contribution of input factors in dynamic models. Reliability Engineering & System Safety. 2011;96:450-9.

[29] Li M, Wang R-Q, Jia G. Efficient dimension reduction and surrogate-based sensitivity analysis for expensive models with high-dimensional outputs. Reliability Engineering & System Safety. 2020;195:106725.

[30] Nagel JB, Rieckermann J, Sudret B. Principal component analysis and sparse polynomial chaos expansions for global sensitivity analysis and model calibration: Application to urban drainage simulation. Reliability Engineering & System Safety. 2020;195:106737.

[31] Perrin T, Roustant O, Rohmer J, Alata O, Naulin J, Idier D et al. Functional principal component analysis for global sensitivity analysis of model with spatial output. Reliability Engineering & System Safety. 2021;211:107522.

[32] Arwade SR, Moradi M, Louhghalam A. Variance decomposition and global sensitivity for structural systems. Engineering Structures. 2010;32:1-10.

[33] Chen W, Jin RC, Sudjianto A. Analytical Variance-Based Global Sensitivity Analysis in Simulation-Based Design Under Uncertainty. Journal of Mechanical Design. 2005;127:875-86.

[34] Zhang X, Pandey MD. An effective approximation for variance-based global sensitivity analysis. Reliability Engineering & System Safety. 2014;121:164-74.

[35] Homma T, Saltelli A. Importance measures in global sensitivity analysis of nonlinear models. Reliability Engineering & System Safety. 1996;52:1-7.

[36] Mara TA, Tarantola S. Variance-based sensitivity indices for models with dependent inputs. Reliability Engineering & System Safety. 2012;107:115-21.

[37] Kucherenko S, Tarantola S, Annoni P. Estimation of global sensitivity indices for models with dependent variables. CoPhC. 2012;183:937-46.

[38] Mara TA, Becker WE. Polynomial chaos expansion for sensitivity analysis of model output with dependent inputs. Reliability Engineering & System Safety. 2021;214:107795.

[39] Gamboa F, Janon A, Klein T, Lagnoux A. Sensitivity analysis for multidimensional and functional outputs. Electronic Journal of Statistics. 2014;8:575-603.

[40] Jolliffe IT. Principal component analysis. 2nd ed. New York: Springer; 2002.

[41] Tipping ME, Bishop CM. Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 1999;61:611-22.

[42] Moon TK. The expectation-maximization algorithm. ISPM. 1996;13:47-60.

[43] McNicholas PD, Murphy TB. Parsimonious Gaussian mixture models. StCom. 2008;18:285-96.

[44] Bishop CM. Pattern recognition and machine learning. New York, NY: Springer; 2006.

[45] Ohtori Y, Christenson R, Spencer Jr B, Dyke S. Benchmark control problems for seismically excited nonlinear buildings. Journal of Engineering Mechanics. 2004;130:366-85.

[46] Patsialis D, Taflanidis A. Reduced order modeling of hysteretic structural response and applications to seismic risk assessment. Engineering Structures. 2020;209:110135.

[47] Kyprioti AP, Adeli E, Taflanidis AA, Westerink JJ, Tolman HL. Probabilistic Storm Surge Estimation for Landfalling Hurricanes: Advancements in Computational Efficiency Using Quasi-Monte Carlo Techniques. Journal of Marine Science and Engineering. 2021;9:1322.

[48] Taylor AA, Glahn B. Probabilistic guidance for hurricane storm surge. 19th Conference on probability and statistics2008.

[49] Kyprioti AP, Taflanidis AA, Nadal-Caraballo NC, Campbell M. Storm hazard analysis over extended geospatial grids utilizing surrogate models. Coastal Engineering. 2021:103855.

[50] Nadal-Caraballo NC, Campbell MO, Gonzalez VM, Torres MJ, Melby JA, Taflanidis AA. Coastal Hazards System: A Probabilistic Coastal Hazard Analysis Framework. Journal of Coastal Research. 2020;95:1211-6.

[51] Luettich RA, Jr. , Westerink JJ, Scheffner NW. ADCIRC: An advanced three-dimensional circulation model for shelves, coasts, and estuaries. Report 1. Theory and methodology of ADCIRC-2DDI and ADCIRC-3DL. Vicksburg,MS: Dredging Research Program Technical Report DRP-92-6, U.S Army Engineers Waterways Experiment Station; 1992.