International Journal for Uncertainty Quantification, 12(2):1–18 (2022)

A COMPREHENSIVE COMPARISON OF TOTAL-ORDER ESTIMATORS FOR GLOBAL SENSITIVITY ANALYSIS

Arnald Puy,[1,*] William Becker,[2] Samuele Lo Piano,[3] & Andrea Saltelli[4]

[1]*Department of Ecology and Evolutionary Biology, Princeton University, Princeton, New Jersey; Centre for the Study of the Sciences and the Humanities (SVT), University of Bergen, Bergen, Norway*

[2]*European Commission, Joint Research Centre, Ispra VA, Italy*

[3]*School of the Built Environment, University of Reading, Reading, UK*

[4]*Barcelona School of Management, Pompeu Fabra University, Barcelona, Spain*

*Address all correspondence to: Arnald Puy, Department of Ecology and Evolutionary Biology, Princeton University, M31 Guyot Hall, Princeton, NJ 08544, E-mail: apuy@princeton.edu

Original Manuscript Submitted: 2/21/2021; Final Draft Received: 7/3/2021

Sensitivity analysis helps identify which model inputs convey the most uncertainty to the model output. One of the most authoritative measures in global sensitivity analysis is the Sobol' total-order index, which can be computed with several different estimators. Although previous comparisons exist, it is hard to know which estimator performs best since the results are contingent on the benchmark setting defined by the analyst (the sampling method, the distribution of the model inputs, the number of model runs, the test function or model and its dimensionality, the weight of higher order effects, or the performance measure selected). Here we compare several total-order estimators in an eight-dimension hypercube, where these benchmark parameters are treated as random parameters. This arrangement significantly relaxes the dependency of the results on the benchmark design. We observe that the most accurate estimators are from Razavi and Gupta, Jansen, or Janon/Monod for factor prioritization, and from Jansen, Janon/Monod, or Azzini and Rosati for approaching the "true" total-order indices. The rest lag considerably behind. Our work helps analysts navigate myriad total-order formulae by reducing the uncertainty in the selection of the most appropriate estimator.

KEY WORDS: *uncertainty analysis, sensitivity analysis, modeling, Sobol' indices, variance decomposition, benchmarking analysis*

1. INTRODUCTION

Sensitivity analysis (SA), i.e., the assessment of how much uncertainty in a given model output is conveyed by each model input, is a fundamental step to judge the quality of model-based inferences [1–3]. Among the many sensitivity indices available, variance-based indices are widely regarded as the gold standard because they are model-free (no assumptions are made about the model), global (they account for interactions between the model inputs), and easy to interpret [4–6]. Given a model of the form $y = f(\boldsymbol{x})$, $\boldsymbol{x} = (x_1, x_2, ..., x_i, ..., x_k) \in \mathbb{R}^k$, where y is a scalar output and $x_1, ..., x_k$ are the k-independent model inputs, the variance of y is decomposed into conditional terms as follows:

$$V(y) = \sum_{i=1}^{k} V_i + \sum_{i} \sum_{i<j} V_{ij} + ... + V_{1,2,...,k} \tag{1}$$

where

$$V_i = V_{x_i}\left[E_{\boldsymbol{x}_{\sim i}}(y|x_i)\right], \quad V_{ij} = V_{x_i,x_j}\left[E_{\boldsymbol{x}_{\sim i,j}}(y|x_i,x_j)\right] - V_{x_i}\left[E_{\boldsymbol{x}_{\sim i}}(y|x_i)\right] - V_{x_j}\left[E_{\boldsymbol{x}_{\sim j}}(y|x_j)\right] \tag{2}$$

and so on up to the kth order. The notation $\boldsymbol{x}_{\sim i}$ means all-but-x_i. By dividing each term in Eq. (1) by the unconditional model output variance $V(y)$, we obtain the first-order indices for single inputs (S_i), pairs of inputs (S_{ij}), and for all higher order terms. First-order indices thus provide the proportion of $V(y)$ caused by each term and are widely used to rank model inputs according to their contribution to the model output uncertainty, a setting known as factor prioritization [1].

Homma and Saltelli [7] also proposed the calculation of the total-order index T_i, which measures the first-order effect of a model input jointly with its interactions up to the kth order

$$T_i = 1 - \frac{V_{\boldsymbol{x}_{\sim i}}\left[E_{x_i}(y|\boldsymbol{x}_{\sim i})\right]}{V(y)} = \frac{E_{\boldsymbol{x}_{\sim i}}\left[V_{x_i}(y|\boldsymbol{x}_{\sim i})\right]}{V(y)} \tag{3}$$

When $T_i \approx 0$, it can be concluded that x_i has a negligible contribution to $V(y)$. For this reason, total-order indices have been applied to distinguish influential from noninfluential model inputs and reduce the dimensionality of the uncertain space, a setting known as factor-fixing [1].

The most direct computation of T_i is via Monte Carlo (MC) estimation, because it does not impose any assumption on the functional form of the response function, unlike metamodeling approaches [8,9]. The Fourier Amplitude Sensitivity Test (FAST) may also be used to calculate T_i, which involves transforming input variables into periodic functions of a single frequency variable, sampling the model, and analyzing the sensitivity of input variables using Fourier analysis in the frequency domain [10,11]. Although an innovative approach, FAST is sensitive to the characteristic frequencies assigned to input variables and is not a very intuitive method. For these reasons, it has mostly been superseded by MC approaches, or by metamodels when computational expense is a serious issue. In this work we focus on the former.

MC methods require generating a $(N, 2k)$ base sample matrix with either random or quasi-random numbers (e.g., Latin Hypercube Sampling, Sobol' quasi-random numbers [12,13]), where each row is a sampling point and each column a model input. The first k columns are allocated to an $\boldsymbol{A}$ matrix and the remaining k columns to a $\boldsymbol{B}$ matrix, which are known as the base sample matrices. Any point in either $\boldsymbol{A}$ or $\boldsymbol{B}$ can be indicated as x_{vi}, where v and i, respectively, index the row (from 1 to N) and the column (from 1 to k). Then, k additional $\boldsymbol{A}_{\boldsymbol{B}}^{(i)}$ ($\boldsymbol{B}_{\boldsymbol{A}}^{(i)}$) matrices are created, where all columns come from $\boldsymbol{A}$ ($\boldsymbol{B}$) except the ith column, which comes from $\boldsymbol{B}$ ($\boldsymbol{A}$). The numerator in Eq. (3) is finally estimated using the model evaluations obtained from the $\boldsymbol{A}$ ($\boldsymbol{B}$) and $\boldsymbol{A}_{\boldsymbol{B}}^{(i)}$ ($\boldsymbol{B}_{\boldsymbol{A}}^{(i)}$) matrices. Some estimators may also use one-third or $\boldsymbol{X}$ base sample matrices (i.e., $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \ldots, \boldsymbol{X}$); although, the use of more than three matrices has been recently proven inefficient by Lo Piano et al. [14].

1.1 Total-Order Estimators and Uncertainties in the Benchmark Settings

The search for efficient and robust total-order estimators is an active field of research [1,7,15–20]. Although some works have compared their asymptotic properties (i.e., [16]), most studies have promoted empirical comparisons where different estimators are benchmarked against known test functions and specific sample sizes. However valuable these empirical studies may be, Becker [21] observed that the results are very much conditional on the choice of model, its dimensionality, and the selected number of model runs. It is hard to say from previous studies whether an estimator outperforming another truly reflects its higher accuracy or simply its better performance under the narrow statistical design of the study. We extend the list of factors that Becker [21] regards as influential in a given benchmarking exercise and discuss how they affect the relative performance of sensitive estimators, as follows:

- *Sampling Method:* The creation of the base sample matrices can be done using the MC or quasi MC (QMC) methods [12,13]. Compared to the MC method, the QMC method allows one to more effectively map the input space as it leaves smaller unexplored volumes (Fig. S1). However, Kucherenko et al. [22] observed that MC methods might help obtain more accurate sensitivity indices when the model under examination has important high-order terms. Both the MC and QMC methods have been used when benchmarking sensitivity indices [15,23].
- *Form of the Test Function:* Some of the most commonly used functions in a sensitivity analysis are the Ishigami and Homma [24], the Sobol' G and its variants [23,25], the Bratley and Fox [26] or the set of

functions presented in Kucherenko et al. [14,16,18,22,23]. Despite being analytically tractable, these functions capture only one possible interval of model behavior, and the effects of nonlinearities and nonadditivities is typically unknown in real models. This black box nature of models has become more of a concern in the last decades due to the increase in computational power and code complexity (which prevents the analyst from intuitively grasping the model's behavior [27]), and to the higher demand for model transparency [3,28,29]. This renders the functional form of the model similar to a random variable [21], something not accounted for by previous works [14,16,18,23].

- *Function Dimensionality:* Many studies focus on low-dimensional problems, either by using test functions that only require a few model inputs (e.g., the Ishigami function, where $k = 3$), or by using test functions with a flexible dimensionality, but setting k at a small value of, e.g., $k \leq 8$ (Sobol' G [25] or Bratley and Fox [26] functions). This approach trades computational manageability for comprehensiveness by neglecting higher dimensions. It is difficult to tell which estimator might work best in models with tens or hundreds of parameters. Examples of such models can be readily found in the Earth and Environmental Sciences domain [30], including the Soil and Water Assessment Tool (SWAT) model, where $k = 50$ [31], or the Modélisation Environmentale-Surface et Hydrologie (MESH) model, where $k = 111$ [32].

- *Distribution of the Model Inputs:* The large majority of benchmarking exercises assume uniformly-distributed inputs $p(\boldsymbol{x}) \in U(0,1)^k$ [14,16,23,33]. However, there is evidence that the accuracy of T_i estimators might be sensitive to the underlying model input distributions, to the point of overturning the model input ranks [34,35]. Furthermore, in uncertainty analysis (e.g., in decision theory), the analysts may use distributions with peaks for the most likely values derived, for instance, from an expert's elicitation stage.

- *Number of Model Runs:* Sensitivity test functions are generally not computationally expensive and can be run without much concern for computational time. This is frequently not the case for real models, whose high dimensionality and complexity might set a constraint on the total number of model runs available. Under such restrictions, the performance of the estimators of the total-order index depends on their efficiency (how accurate they are given the budget of runs that can be allocated to each model input). There are no specific guidelines as to which total-order estimator might work best under these circumstances [21].

- *Performance Measure Selected:* Typically, a sensitivity estimator has been considered to outperform the rest if, on average, it displays a smaller mean absolute error (MAE), computed as follows:

$$\text{MAE} = \frac{1}{p}\sum_{v=1}^{p}\left(\frac{\sum_{i=1}^{k}|T_i - \hat{T}_i|}{k}\right) \tag{4}$$

where p is the number of replicas of the sample matrix, and T_i and $\hat{T}_i$ are the analytical and the estimated total-order index of the ith input. The MAE is appropriate when the aim is to assess which estimator better approaches the true total-order indices, because it averages the error for both influential and noninfluential indices. However, the analyst might be more interested in using the estimated indices $\hat{\boldsymbol{T}} = \{\hat{T}_1, \hat{T}_2, ..., \hat{T}_i, ..., \hat{T}_k\}$ to accurately rank parameters or screen influential from noninfluential model inputs [1]. In such context, the MAE may be best substituted or complemented with a measure of rank concordance between the vectors $\boldsymbol{r}$ and $\hat{\boldsymbol{r}}$, which reflect the ranks in $\boldsymbol{T}$ and $\hat{\boldsymbol{T}}$, respectively, such as the Spearman ρ or the Kendall W coefficient [21,36,37]. It can also be the case that disagreements on the exact ranking of low-ranked parameters may have no practical importance because the interest lies in the correct identification of top ranks only [30]. Savage [38] scores or other measures that emphasize this top-down correlation are then a more suitable choice.

Here, we benchmark the performance of eight different MC-based formulae available to estimate T_i (Table 1). Although the list is not exhaustive, it reflects the research conducted on T_i over the last 20 years: from the classic estimators of Homma and Saltelli [7], Jansen [15], and Saltelli et al. [1] up to the new contributions by Janon et al.

TABLE 1: Formulae to compute T_i. f_0 and $V(y)$ are estimated according to the original papers. For estimators 2 and 5, $f_0 = (1/N)\sum_{v=1}^{N} f(\boldsymbol{A})_v$. For estimators 1, 2, and 5, $V(y) = (1/N)\sum_{v=1}^{N}[f(\boldsymbol{A})_v - f_0]^2$ [1, Eq. (4.16)] and [7, Eqs. (15) and (20)]. For estimator 3, $f_0 = (1/N)\sum_{v=1}^{N}\left[f(\boldsymbol{A})_v + f\left(\boldsymbol{A}_B^{(i)}\right)_v\right]/2$ and $V(y) = (1/N)\sum_{v=1}^{N}\left[f(\boldsymbol{A})_v^2 + f\left(\boldsymbol{A}_B^{(i)}\right)_v^2\right]/2 - f_0^2$ [16, Eq. (15)]. In estimator 4, $\langle f(\boldsymbol{A})_v \rangle$ is the mean of $f(\boldsymbol{A})_v$. We use a simplified version of the Glen and Isaacs [17] estimator because spurious correlations are zero by design. As for estimator 7, we refer to it as pseudo-Owen given its use of a $\boldsymbol{C}$ matrix and its identification with Owen [40] in Iooss et al. [41], where we retrieve the formula from. $V(y)$ in estimator 7 is computed as in estimator 3 following Iooss et al. [41], whereas $V(y)$ in estimator 8 is computed as in estimator 1

No.	Estimator	Authors
1	$\dfrac{(1/2N)\sum_{v=1}^{N}\left[f(\boldsymbol{A})_v - f(\boldsymbol{A}_B^{(i)})_v\right]^2}{V(y)}$	Jansen [15]
2	$\dfrac{V(y) - (1/N)\sum_{v=1}^{N} f(\boldsymbol{A})_v f(\boldsymbol{A}_B^{(i)})_v + f_0^2}{V(y)}$	Homma and Saltelli [7]
3	$1 - \dfrac{(1/N)\sum_{v=1}^{N} f(\boldsymbol{A})_v f(\boldsymbol{A}_B^{(i)})_v - f_0^2}{V(y)}$	Janon et al. [16] and Monod et al. [19]
4	$1 - \left[\dfrac{1}{N-1}\sum_{v=1}^{N}\dfrac{[f(\boldsymbol{A})_v - \langle f(\boldsymbol{A})_v\rangle]\left[f(\boldsymbol{A}_B^{(i)})_v - \left\langle f(\boldsymbol{A}_B^{(i)})_v\right\rangle\right]}{\sqrt{V[f(\boldsymbol{A})_v]V\left[f(\boldsymbol{A}_B^{(i)})_v\right]}}\right]$	Glen and Isaacs [17]
5	$1 - \dfrac{(1/N)\sum_{v=1}^{N} f(\boldsymbol{B})_v f(\boldsymbol{B}_A^{(i)})_v - f_0^2}{V(y)}$	Saltelli et al. [1]
6	$\dfrac{\sum_{v=1}^{N}\left[f(\boldsymbol{B})_v - f\left(\boldsymbol{B}_A^{(i)}\right)_v\right]^2 + \left[f(\boldsymbol{A})_v - f(\boldsymbol{A}_B^{(i)})_v\right]^2}{\sum_{v=1}^{N}[f(\boldsymbol{A})_v - f(\boldsymbol{B})_v]^2 + \left[f\left(\boldsymbol{B}_A^{(i)}\right)_v - f\left(\boldsymbol{A}_B^{(i)}\right)_v\right]^2}$	Azzini and Rosati [33] and Azzini et al. [18]
7	$\dfrac{V(y) - \left[(1/N)\sum_{v=1}^{N}\left\{\left[f(\boldsymbol{B})_v - f(\boldsymbol{C}_B^{(i)})_v\right]\left[f\left(\boldsymbol{B}_A^{(i)}\right)_v - f(\boldsymbol{A})_v\right]\right\}\right]}{V(y)}$	pseudo-Owen
8	$\dfrac{E_{x^*_{\sim_i}}[\gamma_{x^*\sim i}(h_i)] + E_{x^*\sim i}[C_{x^*\sim i}(h_i)]}{V(y)}$	Razavi and Gupta [20,39]

[16], Glen and Isaacs [17], Azzini and Rosati [33], and Razavi and Gupta [20,39]. In order to reduce the influence of the benchmarking design in the assessment of the estimators' accuracy, we treat the sampling method τ, the underlying model input distribution ϕ, the number of model runs N_t, the test function ε, its dimensionality and degree of nonadditivity (k, k_2, k_3), and the performance measure δ as random parameters. This better reflects the diversity of models and sensitivity settings available to the analyst. By relaxing the dependency of the results on these benchmark parameters, we define an unprecedentedly large setting where all formulae can prove their accuracy. Note that we refer to the set of benchmarking assumptions as benchmarking parameters or parameters. This is intended to distinguish them from the inputs of each test function generated by the metafunction, which we refer to as inputs.

We therefore extend the Becker [21] approach by testing a wider set of Monte Carlo estimators, by exploring a wider range of benchmarking assumptions and by performing a formal SA on these assumptions. The aim is therefore to provide a much more global comparison of available MC estimators than is available in the existing literature and to investigate how the benchmarking parameters may affect the relative performance of estimators. Such information

can help point to estimators that are not only efficient on a particular case study, but efficient and robust to a wide range of practical situations.

2. ASSESSMENT OF THE UNCERTAINTIES IN THE BENCHMARKING PARAMETERS

In this section, we formulate the benchmarking parameters as random variables and assess how the performance of estimators is dependent on them by performing a SA. In essence, this is a sensitivity analysis of sensitivity analyses [42], and a natural extension of a similar uncertainty analysis in a recent work by Becker [21]. The use of global SA tools to better understand the properties of estimators can give insights into how estimators behave in different scenarios that are not available through analytical approaches.

2.1 The Setting

The variability in the benchmark settings ($\tau, N_t, k, k_2, k_3, \phi, \epsilon, \delta$) is described by probability distributions (Table 2). We assign uniform distributions (discrete or continuous) to each parameter. In particular, we choose $\tau \sim \mathcal{DU}(1,2)$ to check how the performance of T_i estimators is conditioned by the use of MC ($\tau = 1$) or QMC ($\tau = 2$) methods in the creation of the base sample matrices. For $\tau = 2$, we use the Sobol' sequence scrambled according to Owen [43] to avoid repeated coordinates at the beginning of the sequence. The total number of model runs and inputs is respectively described as $N_t \sim \mathcal{DU}(10, 1000)$ and $k \sim \mathcal{DU}(3, 100)$ to explore the performance of the estimators in a wide range of N_t, k combinations. Given the sampling constraints set by the estimators' reliance on either a $\boldsymbol{B}$, $\boldsymbol{B}_A^{(i)}$, $\boldsymbol{A}_B^{(i)}$, or $\boldsymbol{C}_B^{(i)}$ matrices (Table 1), we modify the space defined by (N_t, k) to a nonrectangular domain (we provide more information on this adjustment in Section 2.2).

For ϕ we set $\phi \sim \mathcal{DU}(1,8)$ to ensure an adequate representation of the most common shapes in the $(0,1)^k$ domain. Besides the normal distribution truncated at $(0,1)$ and the uniform distribution, we also take into account four beta distributions parametrized with distinct α and β values and a logitnormal distribution [Fig. 1(a)]. The aim is to check the response of the estimators under a wide range of probability distributions, including U-shaped distributions and distributions with different degrees of skewness.

We link each distribution in Fig. 1(a) to an integer value from 1 to 7. For instance, if $\phi = 1$, the joint probability distribution of the model inputs is described as $p(x_1, \ldots, x_k) = \mathcal{U}(0,1)^k$. If $\phi = 8$, we create a vector $\boldsymbol{\phi} = \{\phi_1, \phi_2, ..., \phi_i, ..., \phi_k\}$ by randomly sampling the seven distributions in Fig. 1(a), and use the ith distribution in the vector to describe the uncertainty of the ith input. This last case examines the behavior of the estimators when several distributions are used to characterize the uncertainty in the model input space.

2.1.1 The Test Function

The parameter ε operationalizes the randomness in the form and execution of the test function. Our test function is an extended version of the Becker [21] metafunction, which randomly combines p univariate functions in a multivariate function of dimension k. Here we consider the 10 univariate functions listed in Fig. 1(b), which represent common

TABLE 2: Summary of the parameters and their distributions. $\mathcal{DU}$ stands for discrete uniform

Parameter	Description	Distribution
τ	Sampling method	$\mathcal{DU}(1,2)$
N_t	Total number of model runs	$\mathcal{DU}(10,1000)$
k	Number of model inputs	$\mathcal{DU}(3,100)$
ϕ	Probability distribution of the model inputs	$\mathcal{DU}(1,8)$
ε	Randomness in the test function	$\mathcal{DU}(1,200)$
k_2	Fraction of pairwise interactions	$\mathcal{U}(0.3,0.5)$
k_3	Fraction of three-wise interactions	$\mathcal{U}(0.1,0.3)$
δ	Selection of the performance measure	$\mathcal{DU}(1,2)$

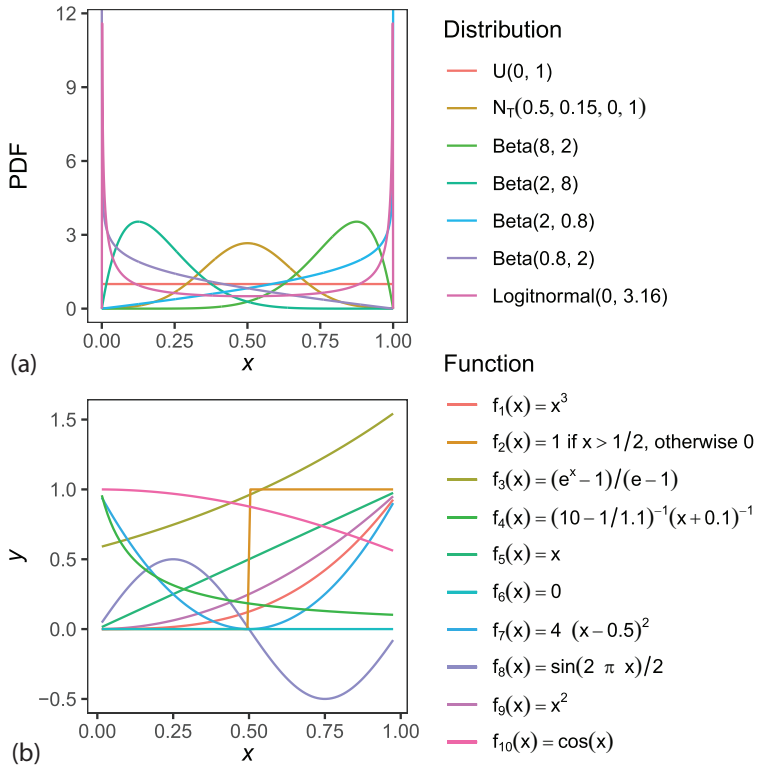

FIG. 1: The metafunction approach. (a) Probability distributions incuded in ϕ. N_T stands for truncated normal distribution. (b) Univariate functions included in the metafunction ($f_1(x) =$ cubic, $f_2(x) =$ discontinuous, $f_3(x) =$ exponential, $f_4(x) =$ inverse, $f_5(x) =$ linear, $f_6(x) =$ no effect, $f_7(x) =$ non-monotonic, $f_8(x) =$ periodic, $f_9(x) =$ quadratic, and $f_{10}(x) =$ trigonometric).

responses observed in physical systems and in classic SA test functions (see Becker [21] for a discussion on this point). We note that an alternative approach would be to construct orthogonal basis functions that could allow analytical evaluation of true sensitivity indices for each generated function; however, this extension is left for future work.

We construct the test function as follows:

1. Let us consider a sample matrix such as

$$\boldsymbol{M} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1i} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2i} & \cdots & x_{2k} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{v1} & x_{v2} & \cdots & x_{vi} & \cdots & x_{vk} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Ni} & \cdots & x_{Nk} \end{bmatrix} \tag{5}$$

   where every point $\boldsymbol{x}_v = x_{v1}, x_{v2}, \ldots, x_{vk}$ represents a given combination of values for the k inputs and x_i is a model input whose distribution is defined by ϕ.

2. Let $\boldsymbol{u} = \{u_1, u_2, ..., u_k\}$ be a k-length vector formed by randomly sampling with replacement of the 10 functions in Fig. 1(b). The ith function in $\boldsymbol{u}$ is then applied to the ith model input; for instance, if $k = 4$ and $\boldsymbol{u} = \{u_3, u_4, u_8, u_1\}$, then $f_3(x_1) = (e^{x_1} - 1)/(e - 1)$, $f_4(x_2) = [10 - (1/1.1)]^{-1}(x_2 + 0.1)^{-1}$, $f_8(x_3) = [\sin(2\pi x_3)]/2$, and $f_1(x_4) = x_4^3$. The elements in $\boldsymbol{u}$ thus represent the first-order effects of each model input.

3. Let $\boldsymbol{V}$ be a $(n, 2)$ matrix, for $n = k!/[2!(k-2)!]$, the number of pairwise combinations between the k inputs of the model. Each row in $\boldsymbol{V}$ thus specifies an interaction between two columns in $\boldsymbol{M}$. In the case of $k = 4$ and the same elements in $\boldsymbol{u}$ as defined in the previous example,

$$\boldsymbol{V} = \begin{bmatrix} 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 2 & 3 \\ 2 & 4 \\ 3 & 4 \end{bmatrix} \tag{6}$$

e.g., the first row promotes $f_3(x_1) \cdot f_4(x_2)$, the second row $f_3(x_1) \cdot f_8(x_3)$, and so on until the nth row. In order to follow the sparsity of effects principle (most variations in a given model output should be explained by low-order interactions [44]), the metafunction activates only a fraction of these effects: it randomly samples $\lceil k_2 n \rceil$ rows from $\boldsymbol{V}$, and computes the corresponding interactions in $\boldsymbol{M}$. $\lceil k_2 n \rceil$ is thus the number of pairwise interactions present in the function. We make k_2 an uncertain parameter described as $k_2 \sim \mathcal{U}(0.3, 0.5)$ in order to randomly activate only between 30 and 50% of the available second-order effects in $\boldsymbol{M}$.

4. Same as before, but for third-order effects: let $\boldsymbol{W}$ be a $(m, 3)$ matrix, for $m = k!/[3!(k-3)!]$, the number of three-wise combinations between the k inputs in $\boldsymbol{M}$. For $k = 4$ and $\boldsymbol{u}$ as before,

$$\boldsymbol{W} = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 4 \\ 1 & 3 & 4 \\ 2 & 3 & 4 \end{bmatrix} \tag{7}$$

e.g., the first row leads to $f_3(x_1) \cdot f_4(x_2) \cdot f_8(x_3)$, and so on until the mth row. The metafunction then randomly samples $\lceil k_3 m \rceil$ rows from $\boldsymbol{W}$ and computes the corresponding interactions in $\boldsymbol{M}$. $\lceil k_3 m \rceil$ is therefore the number of three-wise interaction terms in the function. We also make k_3 an uncertain parameter described as $k_3 \sim \mathcal{U}(0.1, 0.3)$ to activate only between 10 and 30% of all third-order effects in $\boldsymbol{M}$. Note that $k_2 > k_3$ because third-order effects tend to be less dominant than two-order effects (Table 2).

5. Three vectors of coefficients $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ of length k, n and m are defined to represent the weights of the first-, second-, and third-order effects, respectively. These coefficients are generated by sampling from a mixture of two normal distributions $\Psi = 0.3\mathcal{N}(0, 5) + 0.7\mathcal{N}(0, 0.5)$. This coerces the metafunction into replicating the Pareto [45] principle (around 80% of the effects are due to 20% of the parameters), found to widely apply in SA [1,46].

6. The metafunction can thus be formalized as follows:

$$\begin{aligned} y = &\sum_{i=1}^{k} \alpha_i f^{u_i} \phi_i(x_i) + \sum_{i=1}^{\lceil k_2 n \rceil} \beta_i f^{u_{V_{i,1}}} \phi_i(x_{V_{i,1}}) f^{u_{V_{i,2}}} \phi_i(x_{V_{i,2}}) \\ &+ \sum_{i=1}^{\lceil k_3 m \rceil} \gamma_i f^{u_{W_{i,1}}} \phi_i(x_{W_{i,1}}) f^{u_{W_{i,2}}} \phi_i(x_{W_{i,2}}) f^{u_{W_{i,3}}} \phi_i(x_{W_{i,3}}) \end{aligned} \tag{8}$$

Note that there is randomness in the sampling of $\boldsymbol{\phi}$, the univariate functions in $\boldsymbol{u}$, and the coefficients in $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$. The parameter ε assesses the influence of this randomness by fixing the starting point of the pseudo-random number sequence used for sampling the parameters just mentioned. We use $\varepsilon \sim \mathcal{U}(1, 200)$ to ensure that the same seed does not overlap with the same value of N_t, k or any other parameter, an issue that might introduce determinism in a process that should be stochastic. In Figs. S2 and S3 we show the type of T_i indices generated by this metafunction.

Finally, we describe the parameter δ as $\delta \sim \mathcal{DU}(1,2)$. If $\delta = 1$, we compute the Kendall τ-b correlation coefficient between $\hat{\boldsymbol{r}}$ and $\boldsymbol{r}$, the estimated and the "true" ranks calculated from $\hat{\boldsymbol{T}}$ and $\boldsymbol{T}$, respectively. This aims at evaluating how well the estimators in Table 1 rank all model inputs. If $\delta = 2$, we compute the Pearson correlation between $\boldsymbol{r}$ and $\hat{\boldsymbol{r}}$ after transforming the ranks to Savage scores [38]. This setting examines the performance of the estimators when the analyst is interested in ranking only the most important model inputs. Savage scores are given as follows:

$$\mathrm{Sa}_i = \sum_{j=i}^{k} \frac{1}{j} \tag{9}$$

where j is the rank assigned to the jth element of a vector of length k. If $x_1 > x_2 > x_3$, the Savage scores (Sa) would then be $\mathrm{Sa}_1 = 1 + (1/2) + (1/3)$, $\mathrm{Sa}_2 = (1/2) + (1/3)$, and $\mathrm{Sa}_3 = 1/3$. The parameter δ thus assesses the accuracy of the estimators in properly ranking the model inputs, in other words, when they are used in a factor prioritization setting [1].

In order to also examine how accurate the estimators are in approaching the true indices, we ran an extra round of simulations with the MAE as the only performance measure, which we compute as follows:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{k} |T_i - \hat{T}_i|}{k} \tag{10}$$

Note that, unlike Eq. (4), Eq. (10) does not make use of replicas. This is because the effect of the sampling is averaged out in our design by simultaneously varying all parameters in many different simulations.

2.2 Execution of the Algorithm

We examine how sensitive the performance of total-order estimators is to the uncertainty in the benchmark parameters $\tau, N_t, k, k_2, k_3, \phi, \epsilon, \delta$ by means of a global SA. We create $\boldsymbol{A}$, $\boldsymbol{B}$, and $k-1$ $\boldsymbol{A}_B^{(i)}$ matrices, each of dimension $(2^{11}, k)$, using Sobol' quasi-random numbers. In these matrices each column is a benchmark parameter described with the probability distributions of Table 2 and each row is a simulation with a specific combination of $\tau, N_t, k, \ldots$ values. Note that we use $k-1$ $\boldsymbol{A}_B^{(i)}$ matrices because we group N_t and k and treat them like a single benchmark parameter given their correlation (see the list that follows).

Our algorithm runs rowwise over the $\boldsymbol{A}$, $\boldsymbol{B}$, and $k-1$ $\boldsymbol{A}_B^{(i)}$ matrices, for $v = 1, 2, \ldots,$ 18,432 rows. In the vth row, it does the following:

1. It creates five (N_{t_v}, k_v) matrices using the sampling method defined by τ_v. The need for these five submatrices responds to the five specific sampling designs requested by the estimators of our study (Table 1). We use these matrices to compute the vector of estimated indices $\hat{\boldsymbol{T}}_i$ for each estimator:

   a. An $\boldsymbol{A}$ matrix and k_v $\boldsymbol{A}_B^{(i)}$ matrices, each of size (N_v, k_v), $N_v = \lceil\lceil N_{t_v}/(k_v+1)\rceil\rceil$ (estimators 1–4 in Table 1).

   b. An $\boldsymbol{A}$, $\boldsymbol{B}$, and k_v $\boldsymbol{A}_B^{(i)}$ matrices, each of size (N_v, k_v), $N_v = \lceil\lceil N_{t_v}/(k_v+2)\rceil\rceil$ (estimator 5 in Table 1).

   c. An $\boldsymbol{A}$, $\boldsymbol{B}$, and k_v $\boldsymbol{A}_B^{(i)}$ and $\boldsymbol{B}_A^{(i)}$ matrices, each of size (N_v, k_v), $N_v = \lceil\lceil N_{t_v}/(2k_v+2)\rceil\rceil$ (estimator 6 in Table 1).

   d. An $\boldsymbol{A}$, $\boldsymbol{B}$, and k_v $\boldsymbol{B}_A^{(i)}$ and $\boldsymbol{C}_B^{(i)}$ matrices, each of size (N_v, k_v), $N_v = \lceil\lceil N_{t_v}/(2k_v+2)\rceil\rceil$ (estimator 7 in Table 1).

   e. A matrix formed by N_v stars, each of size $k_v(1/\Delta h - 1) + 1$. Given that we set Δh at 0.2 (see Supplement Section), $N_v = \lfloor\lfloor N_{t_v}/(4k+1)\rfloor\rfloor$ (estimator 8 in Table 1).

   The different sampling designs and the value for k_v constrains the total number of runs N_{t_v} that can be allocated to each estimator. Furthermore, given the probability distributions selected for N_t and k (Table 2),

specific combinations of (N_{t_v}, k_v) lead to $N_v \leq 1$, which is computationally unfeasible. To minimize these issues, we force the comparison between estimators to approximate the same N_{t_v} value. Since the sampling design structure of Razavi and Gupta [20,39] is the most constraining, we use $N_v = 2(4k+1)/(k+1)$ (for estimators 1–4), $N_v = 2(4k+1)/(k+2)$ (for estimator 5), and $N_v = 2(4k+1)/(2k+2)$ (for estimators 6 and 7) when $N_v \leq 1$ in the case of Razavi and Gupta [20,39]. This compels all estimators to explore a very similar portion of the (N_t, k) space, but N_t and k become correlated, which contradicts the requirement of independent inputs characterizing variance-based sensitivity indices [1]. This is why we treat (N_t, k) as a single benchmark parameter in the SA.

2. It creates a sixth matrix, formed by an $\boldsymbol{A}$ and k_v $\boldsymbol{A}_B^{(i)}$ matrices, each of size $(2^{11}, k_v)$. We use this submatrix to compute the vector of true indices $\boldsymbol{T}$, which could not be calculated analytically due to the wide range of possible functional forms created by the metafunction. Following Becker [21], we assume that a fairly accurate approximation to $\boldsymbol{T}$ could be achieved with a large Monte Carlo estimation.

3. The distribution of the model inputs in these six sample matrices is defined by ϕ_v.

4. The metafunction runs over these six matrices simultaneously, with its functional form, and degree of active second- and third-order effects as set by ε_v, k_{2_v}, and k_{3_v}, respectively.

5. It computes the estimated sensitivity indices $\hat{\boldsymbol{T}}_v$ for each estimator and the true sensitivity indices $\boldsymbol{T}_v$ using the Jansen [15] estimator, which is currently best practice in SA.

6. It checks the performance of the estimators. This is done in two ways:

   a. If $\delta = 1$, we compute the correlation between $\hat{\boldsymbol{r}}_v$ and $\boldsymbol{r}_v$ (obtained respectively from $\hat{\boldsymbol{T}}_v$ and $\boldsymbol{T}_v$) with Kendall tau, and if $\delta = 2$, we compute the correlation between $\hat{\boldsymbol{r}}_v$ and $\boldsymbol{r}_v$ on Savage scores. The model output in both cases is the correlation coefficient r, with higher r values indicating a better performance in properly ranking the model inputs.

   b. We compute the MAE between $\hat{\boldsymbol{T}}_v$ and $\boldsymbol{T}_v$. In this case, the model output is the MAE, with lower values indicating a better performance in approaching the true total-order indices.

3. RESULTS

3.1 Uncertainty Analysis

Under a factor prioritization setting (e.g., when the aim is to rank the model inputs in terms of their contribution to the model output variance), the most accurate estimators are Jansen [15], Razavi and Gupta [20,39], Janon/Monod [16,19], and Azzini and Rosati [18,33]. The distribution of r values (the correlation between estimated and true ranks) when these estimators are used is highly negatively skewed, with median values of ≈ 0.9. Glen and Isaacs [17], Homma and Saltelli [7], Saltelli [1], and pseudo-Owen lag behind and display median r values of ≈ 0.35, with pseudo-Owen ranking last ($r \approx 0.2$). The range of values obtained with these formulae is much more spread out and include a significant number of negative r values, suggesting that they overturned the true ranks in several simulations [Figs. 2(a) and S4].

When the goal is to approximate the true indices, Janon/Monod [16,19], Jansen [15], and Azzini and Rosati [18,33] also offer the best performance. The median MAE obtained with these estimators is generally smaller than Glen and Isaacs [17] and pseudo-Owen, and the distribution of MAE values is much narrower than that obtained with Homma and Saltelli [7], Saltelli [1], or Razavi and Gupta [20,39]. These three estimators are the least accurate and produce several MAE values larger than 10^2 in several simulations [Fig. 2(b)]. The volatility of Razavi and Gupta [20,39] under the MAE is reflected in the numerous outliers produced and sharply contrasts with its very good performance in a factor prioritization setting [Fig. 2(a)].

To obtain a finer insight into the structure of these results, we plot the total number of model runs N_t against the function dimensionality k (Fig. 3). This maps the performance of the estimators in the input space formed by all

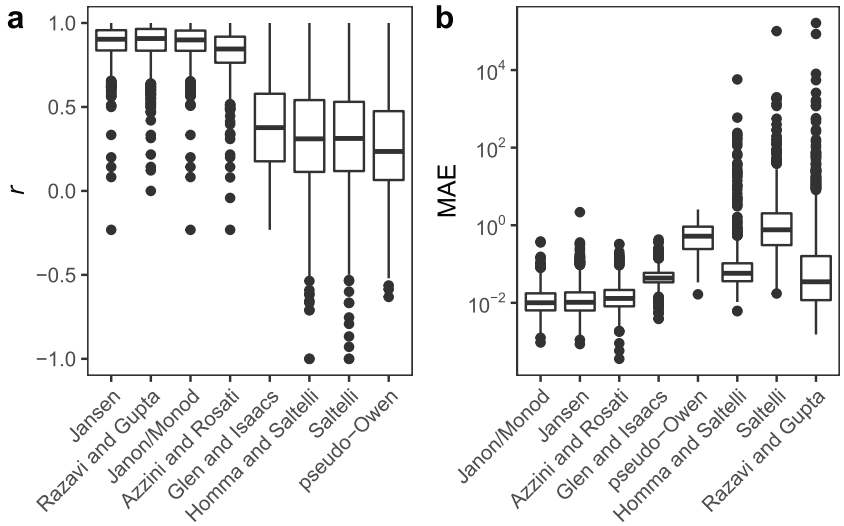

FIG. 2: Boxplots summarizing the results of the simulations. (a) Correlation coefficient between $\hat{\boldsymbol{r}}$ and $\boldsymbol{r}$, the vector of estimated and true ranks. (b) Mean absolute error (MAE).

possible combinations of N_t and k given the specific design constraints of each formulae. Under a factor prioritization setting, almost all estimators perform reasonably well at a very small dimensionality ($k \leq 10, r > 0.7$), regardless of the total number of model runs available. However, some differences unfold at higher dimensions: Saltelli [1], Homma and Saltelli [7], Glen and Isaacs [17], and especially pseudo-Owen swiftly become inaccurate for $k > 10$, even with large values for N_t. Azzini and Rosati [18,33] display a very good performance overall except in the upper N_t, k boundary, where most of the orange dots concentrate. The estimators of Jansen [15], Janon/Monod [16,19], and Razavi and Gupta [20,39] rank the model inputs almost flawlessly regardless of the region explored in the N_t, k domain [Fig. 3(a)].

With regard to the MAE, Janon/Monod [16,19], Jansen [15], and Azzini and Rosati [18,33] maintain their high performance regardless of the N_t, k region explored. The accuracy of Razavi and Gupta [20,39], however, drops at the upper-leftmost part of the N_t, k boundary, where most of the largest MAE scores are located (MAE > 10). In the case of Saltelli [1] and Homma and Saltelli [7], the largest MAE values concentrate in the region of small k regardless of the total number of model runs, a domain in which they achieved a high performance when the focus was on properly ranking the model inputs.

The presence of a non-negligible proportion of model runs with $r < 0$ suggests that some estimators significantly overturned the true ranks [Figs. 3(a) and S4]. To better examine this phenomenon, we re-plot Fig. 3(b) with just the simulations yielding $r < 0$ (Fig. S5). We observe that $r < 0$ values not only appear in the region of small N_t, a foreseeable miscalculation derived from allocating an insufficient number of model runs to each model input: they also emerge at a relatively large N_t and low k in the case of pseudo-Owen, Saltelli [1], and Homma and Saltelli [7]. The Saltelli estimator actually concentrates in the $k < 10$ zone most of the simulations with the lowest negative r values (Fig. S5). This suggests that rank reversing is not an artifact of our study design as much as a by-product of the volatility of these estimators when stressed by the sources of computational uncertainty listed in Table 2. Such strain may lead these estimators to produce a significant fraction of negative indices or indices beyond 1, thus effectively promoting $r < 0$.

We calculate the proportion of $T_i < 0$ and $T_i > 1$ in each simulation that yielded $r < 0$. In the case of Glen and Isaacs [17] and Homma and Saltelli [7], $r < 0$ values are caused by the production of a large proportion of $T_i < 0$ (25–75%, the x-axis in Fig. 4). Pseudo-Owen and Saltelli [1] also suffer this bias, and in several simulations they also generate a large proportion of $T_i > 1$ (up to 100% of the model inputs, the y-axis in Fig. 4). The production of $T_i < 0$ and $T_i > 1$ is caused by numerical errors and fostered by the values generated at the numerator of Eq. (3): $T_i < 0$ may either derive from $E_{\boldsymbol{x}_{\sim i}}\left[V_{x_i}(y|\boldsymbol{x}_{\sim i})\right] < 0$ (e.g., Homma and Saltelli [7] and pseudo-Owen) or

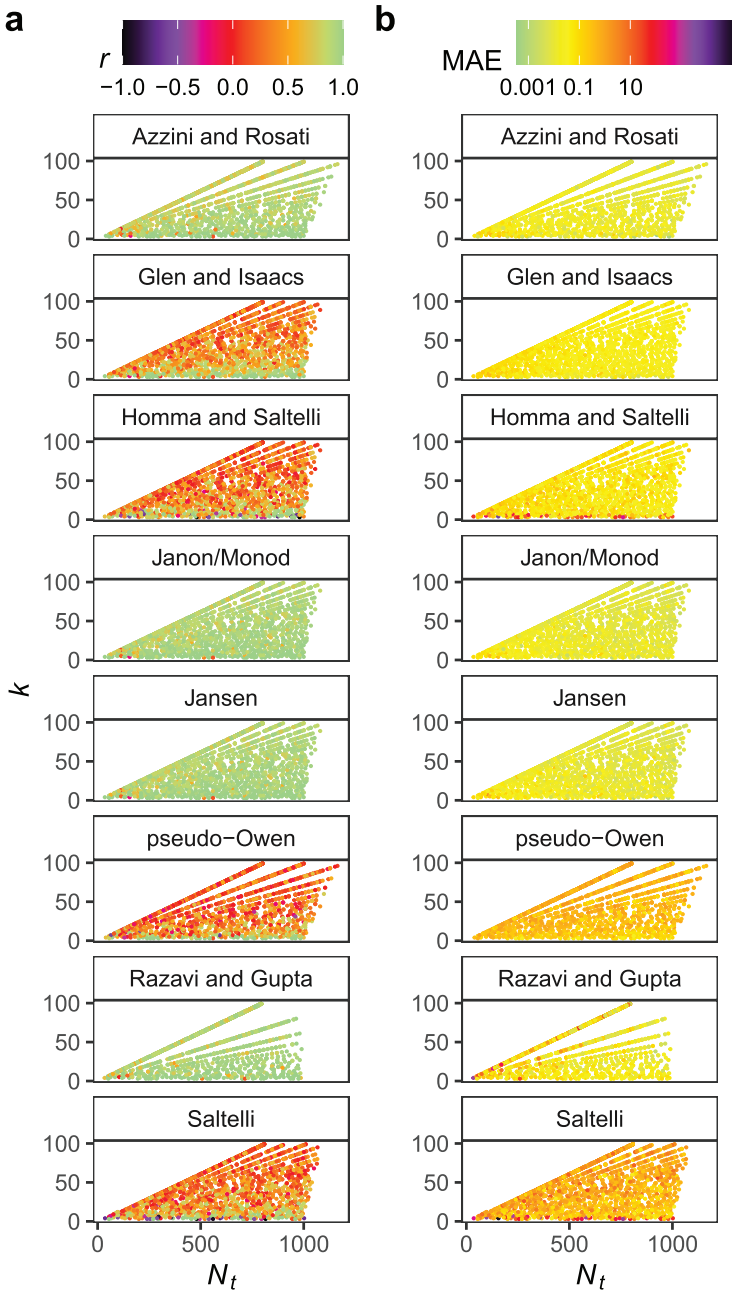

FIG. 3: Number of runs N_t against the function dimensionality k. Each dot is a simulation with a specific combination of the benchmark parameters in Table 2. The greener (blacker) the color is, the better (worse) the performance of the estimator. (a) Accuracy of the estimators when the goal is to properly rank the model inputs, e.g., a factor prioritization setting. (b) Accuracy of the estimators when the goal is to approach the "true" total-order indices.

$V_{\boldsymbol{x}_{\sim i}}\left[E_{x_i}(y|\boldsymbol{x}_{\sim i})\right] > V(y)$ (e.g., Saltelli [1]), whereas $T_i > 1$ from $E_{\boldsymbol{x}_{\sim i}}\left[V_{x_i}(y|\boldsymbol{x}_{\sim i})\right] > V(y)$ (e.g., Homma and Saltelli [7] and pseudo-Owen) or $V_{\boldsymbol{x}_{\sim i}}\left[E_{x_i}(y|\boldsymbol{x}_{\sim i})\right] < 0$ (e.g., Saltelli [1]).

To better examine the efficiency of the estimators, we summarized their performance as a function of the number of runs available per model input N_t/k [21] (Figs. 5 and S6). This information is especially relevant to take an

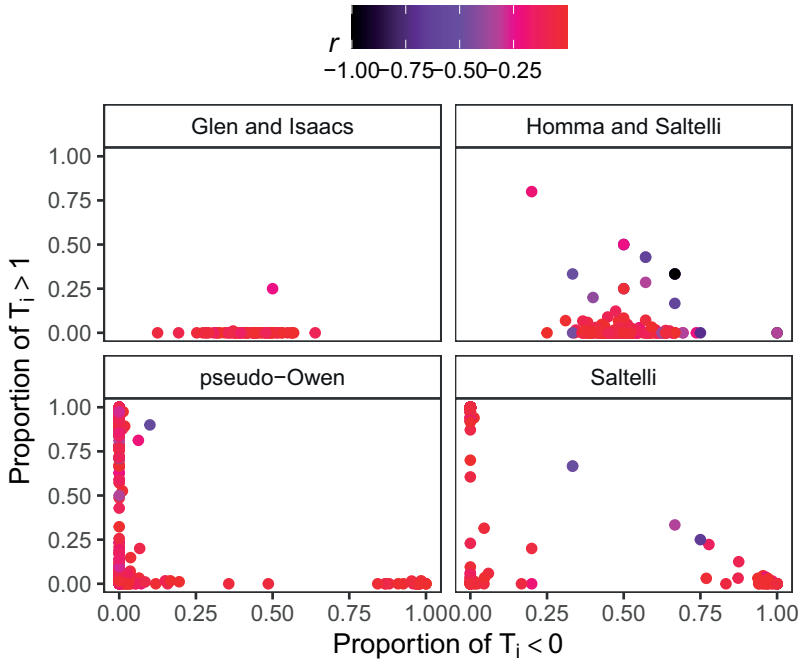

FIG. 4: Scatterplot of the proportion of $T_i < 0$ against the proportion of $T_i > 1$ mapped against the model output r. Each dot is a simulation. Only simulations with $r < 0$ are displayed.

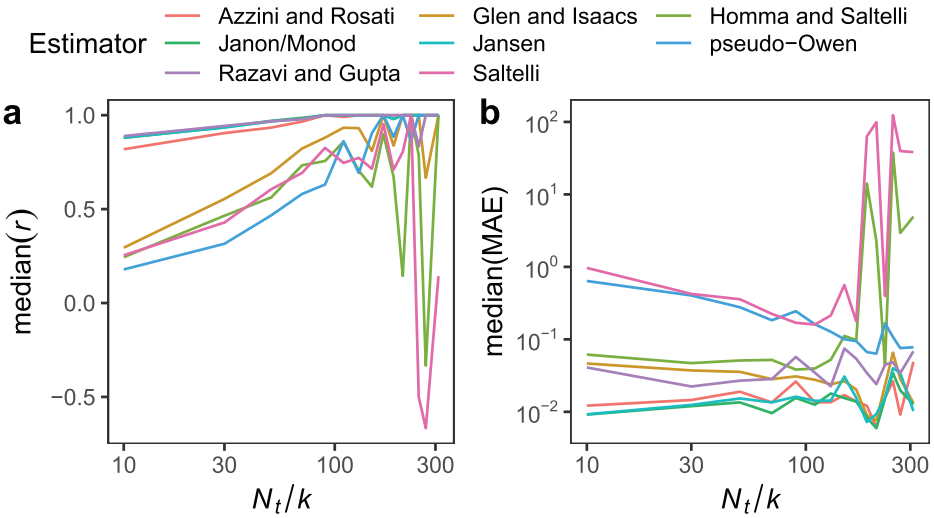

FIG. 5: Scatterplot of the model output r against the number of model runs allocated per model input (N_t/k). See Fig. S6 for a visual display of all simulations and Fig. S7 for an assessment of the number of model runs that each estimator has in each N_t/k compartment.

educated decision on which estimator to use in a context of a high-dimensional, computationally expensive model. Even when the budget of runs per input is low $\{(N_t/k) \in [2, 20]\}$, Razavi and Gupta [20,39], Jansen [15], and Janon/Monod [16,19] are very good at properly ranking model inputs ($r \approx 0.9$), and are followed very close by Azzini and Rosati [18,33] ($r \approx 0.8$). Saltelli [1], Homma and Saltelli [7], and Glen and Isaacs [17] come after ($r \approx 0.3$), with pseudo-Owen scoring last ($r \approx 0.2$). When the N_t/k ratio is increased, all estimators improve their

ranking accuracy and some quickly reach the asymptote: this is the case of Razavi and Gupta [20,39], Janon/Monod [16,19], and Jansen [15], whose performance becomes almost flawless from $(N_t/k) \in [40, 60]$ onward, and of Azzini and Rosati [18,33], which reaches its optimum at $(N_t/k) \in [60, 80]$. The accuracy of the other estimators does not seem to fully stabilize within the range of ratios examined. In the case of Homma and Saltelli [7] and Saltelli [1], their performance oscillates before plummeting at $(N_t/k) \in [200, 210]$, $(N_t/k) \in [240, 260]$ and $(N_t/k) \in [260, 280]$ due to several simulations yielding large $r < 0$ values [Fig. 5(a)].

Janon/Monod [16,19] and Jansen [15] are also the most efficient estimators when the MAE is the measure of choice, followed closely by Azzini and Rosati [18,33], Razavi and Gupta [20,39], and Glen and Isaacs [17]. Saltelli [1] and Homma and Saltelli [7] gain accuracy at higher N_t/k ratios yet their precision diminishes all the same from $(N_t/k) \in [200, 210]$ onward [Fig. 5(b)].

3.2 Sensitivity Analysis

When the aim is to rank the model inputs, the selection of the performance measure (δ) has the highest first-order effect in the accuracy of the estimators [Fig. 6(a)]. The parameter δ is responsible for between 20% (Azzini and Rosati [18,33]) and 30% (Glen and Isaacs [17]) of the variance in the final r value. On average, all estimators perform better when the rank is conducted on Savage scores ($\delta = 2$), i.e., when the focus is on ranking the most important model inputs only (Figs. S8–S15). As for the distribution of the model inputs (ϕ), it has a first-order effect in the accuracy of Azzini and Rosati [18,33] ($\approx 10\%$), Jansen [15], and Janon/Monod [16,19] ($\approx 15\%$) and Razavi and Gupta [20,39] ($\approx 20\%$) regardless of whether the aim is a factor prioritization (r) or approaching the true indices (MAE). The performance of these estimators drops perceptibly when the model inputs are distributed as $Beta(8, 2)$ or $Beta(2, 8)$ ($\phi = 3$ and $\phi = 4$, Figs. S8–S23), suggesting that they may be especially stressed by skewed distributions. The selection of random or quasi-random numbers during the construction of the sample matrix (τ) also directly conditions the accuracy of several estimators. If the aim is to approach the "true" indices (MAE), τ conveys from 17% (Azzini and Rosati [18,33]) to $\approx 30\%$ (Glen and Isaacs [17]) of the model output variance, with all estimators except Razavi and Gupta [20,39] performing better on quasi-random numbers ($\tau = 2$, Figs. S16–S23). In a factor prioritization setting, τ is mostly influential through interactions. Interestingly, the proportion of active second- and third-order interactions (k_2, k_3) does not alter the performance of any estimator in any of the settings examined.

To better understand the structure of the sensitivities, we compute Sobol' indices after grouping individual parameters in three clusters, which we define based on their commonalities: the first group includes (δ, τ) and reflects the influence of those parameters that can be defined by the sensitivity analyst during the setting of the benchmark exercise. The second combines $(\varepsilon, k_2, k_3, \phi)$ and examines the overall impact of the model functional form, referred to as $f(x)$, which is often beyond the analyst's grasp. Finally, the third group includes (N_t, k) only and assesses the influence of the sampling design in the accuracy of the estimators (we assume that the total number of model runs, besides being conditioned by the computing resources at hand, is also partially determined by the joint effect of the model dimensionality and the use of either a $\boldsymbol{B}$, $\boldsymbol{A}_B^{(i)}$, $\boldsymbol{B}_A^{(i)}$, or $\boldsymbol{C}_B^{(i)}$ matrices) [Fig. 6(b)].

The uncertainty in the functional form of the model [$f(x)$] is responsible for $\approx 20\%$ of the variance in the performance of Azzini and Rosati [18,33], Janon/Monod [16,19], or Jansen [15] in a factor prioritization setting. For Glen and Isaacs [17], Homma and Saltelli [7], pseudo-Owen, or Saltelli [1], $f(x)$ is influential only through interactions with the other clusters. When the MAE is the performance measure of interest, $f(x)$ has a much stronger influence in the accuracy of the estimators than the couple (N_t, k), especially in the case of Glen and Isaacs [17] ($\approx 40\%$). In any case, the accuracy of the estimators is significantly conditioned by interactions between the benchmark parameters. The sum of all individual S_i indices plus the S_i index of the (N_t, k) cluster only explains from $\approx 45\%$ (Saltelli [1]) to $\approx 70\%$ (Glen and Isaacs [17]) of the estimators' variance in ranking the model inputs, and from $\approx 24\%$ (pseudo-Owen) to $\approx 60\%$ (Razavi and Gupta [20,39]) of the variance in approaching the true indices.

4. DISCUSSION AND CONCLUSIONS

Here we design an eight-dimension background for variance-based total-order estimators to confront and prove their value in an unparalleled range of SA scenarios. By randomizing the parameters that condition their performance, we

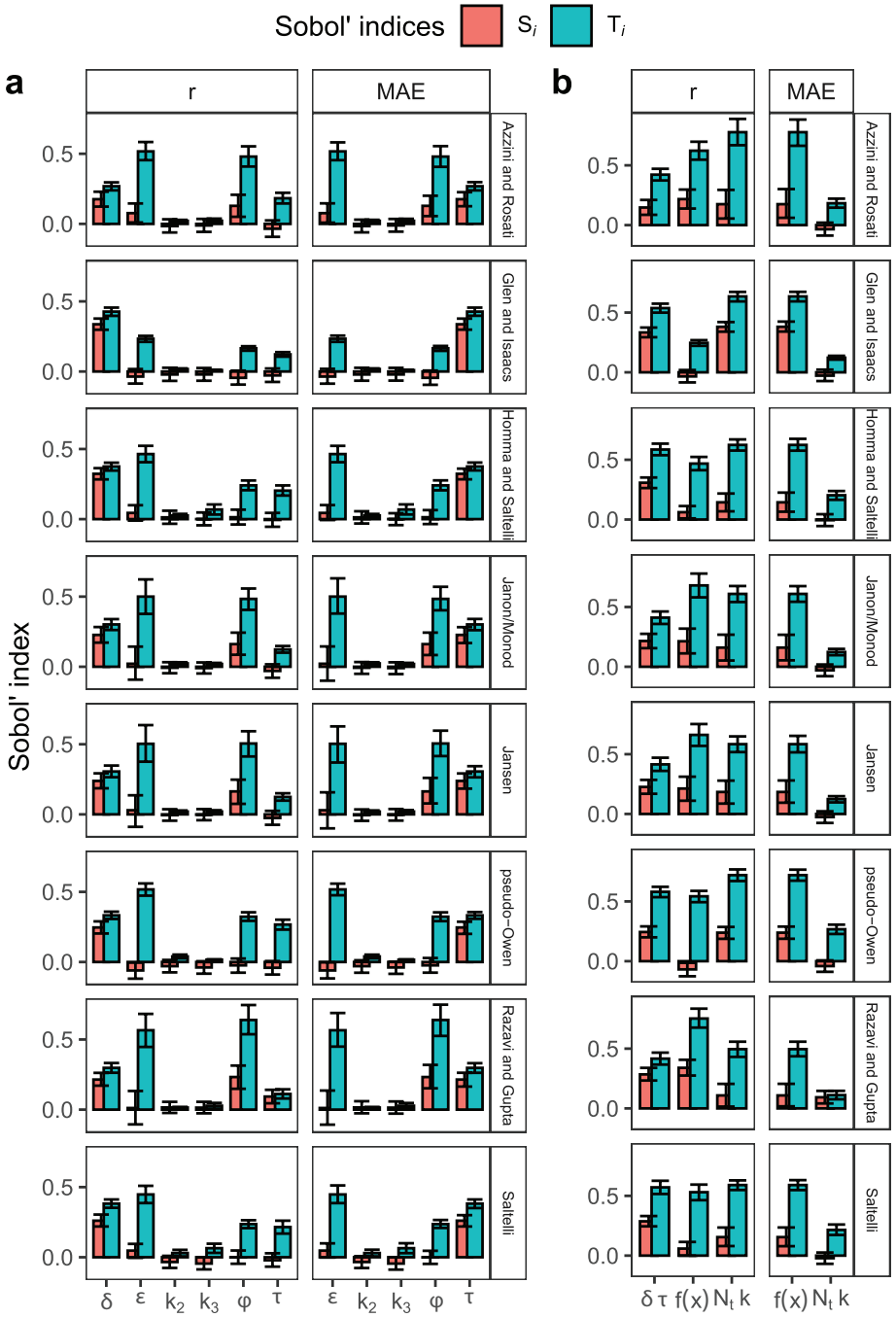

FIG. 6: Sobol' indices. (a) Individual parameters. (b) Clusters of parameters. The cluster $f(x)$ includes all parameters that describe the uncertainty in the functional form of the model (ϵ, k_2, k_3, ϕ). N_t and k are assessed simultaneously due to their correlation. Note that the MAE facet does not include the group ($\delta\tau$) because δ (the performance measure used) is no longer an uncertain parameter in this setting.

obtain a comprehensive picture of the advantages and disadvantages of each estimator and identify which particular benchmark factors make them more prone to error. Our work thus provides a thorough empirical assessment of

state-of-the-art total-order estimators and contributes to define best practices in variance-based SA. The study also aligns with previous works focused on testing the robustness of the tools available to sensitivity analysts, a line of inquiry that can be described as a sensitivity analysis of a sensitivity analysis (SA of SA) [42].

Our results provide support to the assumption that the scope of previous benchmark studies is limited by the plethora of nonunique choices taken during the setting of the analysis [21]. We have observed that almost all decisions have a nonnegligible effect: from the selection of the sampling method to the choice of the performance measure, the design prioritized by the analyst can influence the performance of the estimator in a non-obvious way, namely through interactions. The importance of non-additivities in conditioning performance suggests that the benchmark of sensitivity estimators should no longer rely on statistical designs that change one parameter at a time (usually the number of model runs and, more rarely, the test function [14,16,18,20,23,33,39,40,42]). Such setting reduces the uncertain space to a minimum and misses the effects that the interactions between the benchmark parameters have in the final accuracy of the estimator. If global SA is the recommended practice to fully explore the uncertainty space of models, sensitivity estimators, being algorithms themselves, should be likewise validated [42].

Our approach also compensates the lack of studies on the theoretical properties of estimators in the sensitivity analysis literature (see, for instance, [15,47]), and allows a more detailed examination of their performance than theoretical comparisons. Empirical studies like ours mirror the numerical character of sensitivity analysis when the indices can not be analytically calculated, which is most of the time in real-world mathematical modeling.

Two recommendations emerge from our work: the estimators by Razavi and Gupta [20,39], Jansen [15], Janon/Monod [16,19], or Azzini and Rosati [18,33] should be preferred when the aim is to rank the model inputs. Jansen [15], Janon/Monod [16,19], or Azzini and Rosati [18,33] should also be prioritized if the goal is to estimate the true total-order indices. The drop in performance of Razavi and Gupta [20,39] in the second setting may be explained by a bias at a lower sample sizes, i.e., a consistent overestimation of all total-order indices. This is because their estimator relies on a constant mean assumption whose validity degrades with larger values of Δh [20,39]. In order to remove this bias, Δh should take very small values (e.g., $\Delta h = 0.01$), which may not be computationally feasible. Since the direction of this bias is the same for all parameters it only affects the calculation of the true total-order indices, not the capacity of the estimator to properly rank the model inputs. It is also worth stating that Razavi and Gupta [20,39] is the only estimator studied here that require the analyst to define a tuning parameter, Δh. In this paper, we have set $\Delta h = 0.2$ after some preliminary trials with the estimator; other works have used different values (e.g., $\Delta h = 0.002$, $\Delta h = 0.1$, $\Delta h = 0.3$; see [20,21,39]). Selecting the most appropriate value for a given tuning parameter is not an obvious choice, and this uncertainty can make an estimator volatile, as shown by Puy et al. [42] in the case of the PAWN index.

The fact that Glen and Isaacs [17], Homma and Saltelli [7], Saltelli [1], and pseudo-Owen do not perform as well in properly ranking the model inputs and approaching the true total-order indices may be partially explained by their less efficient computation of elementary effects. By allowing the production of negative terms in the numerator, these estimators also permit the production of negative total-order indices, thus leading to biased rankings or sensitivity indices. In the case of Saltelli [1], the use of a $\boldsymbol{B}$ matrix at the numerator and an $\boldsymbol{A}$ matrix at the denominator exacerbates its volatility (Table 1, estimator 5). Such inconsistency was corrected in Saltelli et al. [23].

The consistent robustness of Jansen [15], Janon/Monod [16,19], and Azzini and Rosati [18,33] makes their sensitivity to the uncertain parameters studied here almost negligible. They are already highly optimized estimators with not much room for improvement. Most of their performance is conditioned by the first- and total-order effects of the model form jointly with the underlying probability distributions [$f(x)$ in Fig. 6(b)], as well as by their sampling design (N_t, k), which are in any case beyond the analyst's control. As for the rest, their accuracy might be enhanced by allocating a larger number of model runs per input (if computationally affordable), and especially in the case of Homma and Saltelli [7], Saltelli [1], and Glen and Isaacs [17], by restricting their use to low-dimensional models ($k < 10$) and sensitivity settings that only require ranking the most important parameters (a restricted factor prioritization setting [1]). Nevertheless, their substantial volatility is considerably driven by non-additivities, a combination that makes them hard to tame and should raise caution about their use in any modeling exercise.

Our results slightly differ from Becker [21], who observed that Jansen [15] outperformed Janon/Monod [16,19] under a factor prioritization setting. We did not find any significant difference between these estimators. Although our metafunction approach is based on Becker [21], our study tests the accuracy of estimators in a larger uncertain space

as we also account for the stress introduced by changes in the sampling method τ, the underlying probability distributions ϕ, or the performance measure selected δ. These differences may account for the slightly different results obtained between the two papers.

Our analysis can be extended to other sensitivity estimators (i.e., moment-independent-like entropy-based [48], the δ-measure [49], or the PAWN index [50,51]). Moreover, it holds potential to be used overall as a standard crash test every time a new sensitivity estimator is introduced to the modeling community. One of its advantages is its flexibility. The Becker [21] metafunction can be easily extended with new univariate functions or probability distributions, and the settings modified to check performance under different degrees of non-additivities or in a larger (N_t, k) space. With some slight modifications it should also allow one to produce functions with dominant low- or high-order terms, labeled as types B and C by Kucherenko et al. [22]. This should prompt developers of sensitivity indices to severely stress their estimators so the modeling community and decision-makers fully appraise how they deal with uncertainties.

5. CODE AVAILABILITY

The `R` code to replicate our results is available in Puy [52]. The uncertainty and sensitivity analysis have been carried out with the `R` package `sensobol` [53], which also includes the test function used in this study.

ACKNOWLEDGMENTS

We thank Saman Razavi for his insights on the Razavi and Gupta estimator. This work has been funded by the European Commission (Marie Skłodowska-Curie Global Fellowship, Grant No. 792178 to A.P.).

REFERENCES

1. Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S., *Global Sensitivity Analysis: The Primer*, Chichester, UK: Wiley, 2008.
2. Jakeman, A., Letcher, R., and Norton, J., Ten Iterative Steps in Development and Evaluation of Environmental Models, *Env. Modell. Software*, **21**(5):602–614, 2006.
3. Eker, S., Rovenskaya, E., Obersteiner, M., and Langan, S., Practice and Perspectives in the Validation of Resource Management Models, *Nat. Commun.*, **9**(1):1–10, 2018.
4. Saltelli, A., Sensitivity Analysis for Importance Assessment, *Risk Anal.*, **22**(3):579–590, 2002.
5. Iooss, B. and Lemaître, P., A Review on Global Sensitivity Analysis Methods, in *Uncertainty Management in Simulation-Optimization of Complex Systems*, Vol. 59, G. Dellino and C. Meloni, Eds., pp. 101–122, Boston: Springer, 2015.
6. Becker, W. and Saltelli, A., Design for Sensitivity Analysis, in *Handbook of Design and Analysis of Experiments*, A. Dean, M. Morris, J. Stufken, and D. Bingham, Eds., Boca Raton, FL: CRC Press, Taylor & Francis, pp. 627–674, 2015.
7. Homma, T. and Saltelli, A., Importance Measures in Global Sensitivity Analysis of Nonlinear Models, *Reliab. Eng. Syst. Saf.*, **52**:1–17, 1996.
8. Le Gratiet, L., Marelli, S., and Sudret, B., Metamodel-Based Sensitivity Analysis: Polynomial Chaos Expansions and Gaussian Processes, in *Handbook of Uncertainty Quantification*, Cham, Switzerland: Springer, pp. 1289–1325, 2017.
9. Saltelli, A., Tarantola, S., and Chan, K.P.S., A Quantitative Model-Independent Method for Global Sensitivity Analysis of Model Output, *Technometrics*, **41**(1):39–56, 1999.
10. Cukier, R.I., Fortuin, C.M., Shuler, K.E., Petschek, A.G., and Schaibly, J.H., Study of the Sensitivity of Coupled Reaction Systems to Uncertainties in Rate Coefficients. I Theory, *The J. Chem. Phys.*, **59**(8):3873–3878, 1973.
11. Cukier, R.I., Levine, H.B., and Shuler, K.E., Nonlinear Sensitivity Analysis of Multiparameter Model Systems, *J. Comput. Phys.*, **26**(1):1–42, 1978.
12. Sobol', I.M., On the Distribution of Points in a Cube and the Approximate Evaluation of Integrals, *USSR Comput. Math. Math. Phys.*, **7**(4):86–112, 1967.
13. Sobol', I.M., Uniformly Distributed Sequences with an Additional Uniform Property, *USSR Comput. Math. Math. Phys.*, **16**(5):236–242, 1976.

14. Lo Piano, S., Ferretti, F., Puy, A., Albrecht, D., and Saltelli, A., Variance-Based Sensitivity Analysis: The Quest for Better Estimators and Designs between Explorativity and Economy, *Reliab. Eng. Syst. Saf.*, **206**:107300, 2021.
15. Jansen, M., Analysis of Variance Designs for Model Output, *Comput. Phys. Commun.*, **117**(1-2):35–43, 1999.
16. Janon, A., Klein, T., Lagnoux, A., Nodet, M., and Prieur, C., Asymptotic Normality and Efficiency of Two Sobol Index Estimators, *ESAIM: Probab. Stat.*, **18**(3):342–364, 2014.
17. Glen, G. and Isaacs, K., Estimating Sobol Sensitivity Indices Using Correlations, *Env. Modell. Software*, **37**:157–166, 2012.
18. Azzini, I., Mara, T., and Rosati, R., Monte Carlo Estimators of First- and Total-Orders Sobol' Indices, *Stat. Appl.*, arXiv:2006.08232, 2020.
19. Monod, H., Naud, C., and Makowski, D., Uncertainty and Sensitivity Analysis for Crop Models, in *Working with Dynamic Crop Models*, D. Wallach, D. Makowski, and J.W. Jones, Eds., New York: Elsevier, pp. 35–100, 2006.
20. Razavi, S. and Gupta, H.V., A New Framework for Comprehensive, Robust, and Efficient Global Sensitivity Analysis: 2. Application, *Water Res. Res.*, **52**(1):440–455, 2016.
21. Becker, W., Metafunctions for Benchmarking in Sensitivity Analysis, *Reliab. Eng. Syst. Saf.*, **204**:107189, 2020.
22. Kucherenko, S., Feil, B., Shah, N., and Mauntz, W., The Identification of Model Effective Dimensions Using Global Sensitivity Analysis, *Reliab. Eng. Syst. Saf.*, **96**(4):440–449, 2011.
23. Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., and Tarantola, S., Variance Based Sensitivity Analysis of Model Output. Design and Estimator for the Total Sensitivity Index, *Comput. Phys. Commun.*, **181**(2):259–270, 2010.
24. Ishigami, T. and Homma, T., An Importance Quantification Technique in Uncertainty Analysis for Computer Models, *Proc. First Int. Symp. Uncert. Modell. Anal.*, **12**:398–403, 1990.
25. Sobol', I.M., On Quasi-Monte Carlo Integrations, *Math. Comput. Simul.*, **47**(2-5):103–112, 1998.
26. Bratley, P. and Fox, B.L., ALGORITHM 659: Implementing Sobol's Quasirandom Sequence Generator, *ACM Trans. Math. Software*, **14**(1):88–100, 1988.
27. Borgonovo, E. and Plischke, E., Sensitivity Analysis: A Review of Recent Advances, *Eur. J. Oper. Res.*, **248**(3):869–887, 2016.
28. Saltelli, A., A Short Comment on Statistical versus Mathematical Modelling, *Nat. Commun.*, **10**(1):8–10, 2019.
29. Saltelli, A., Bammer, G., Bruno, I., Charters, E., Di Fiore, M., Didier, E., Nelson Espeland, W., Kay, J., Lo Piano, S., Mayo, D., Pielke, R., Jr., Portaluri, T., Porter, T.M., Puy, A., Rafols, I., Ravetz, J.R., Reinert, E., Sarewitz, D., Stark, P.B., Stirling, A., van der Sluijs, J., and Vineis, P., Five Ways to Ensure That Models Serve Society: A Manifesto, *Nature*, **582**(7813):482–484, 2020.
30. Sheikholeslami, R., Razavi, S., Gupta, H.V., Becker, W., and Haghnegahdar, A., Global Sensitivity Analysis for High-Dimensional Problems: How to Objectively Group Factors and Measure Robustness and Convergence While Reducing Computational Cost, *Env. Modell. Software*, **111**:282–299, 2019.
31. Sarrazin, F., Pianosi, F., and Wagener, T., Global Sensitivity Analysis of Environmental Models: Convergence and Validation, *Env. Modell. Software*, **79**:135–152, 2016.
32. Haghnegahdar, A., Razavi, S., Yassin, F., and Wheater, H., Multicriteria Sensitivity Analysis as a Diagnostic Tool for Understanding Model Behaviour and Characterizing Model Uncertainty, *Hydrol. Proces.*, **31**(25):4462–4476, 2017.
33. Azzini, I. and Rosati, R., The IA-Estimator for Sobol' Sensitivity Indices, in *9th Int. Conf. on Sensitivity Analysis of Model Output*, Barcelona, Spain, 2019.
34. Shin, M.J., Guillaume, J.H.A., Croke, B.F.W., and Jakeman, A.J., Addressing Ten Questions about Conceptual Rainfall-Runoff Models with Global Sensitivity Analyses in R, *J. Hydrol.*, **503**:135–152, 2013.
35. Paleari, L. and Confalonieri, R., Sensitivity Analysis of a Sensitivity Analysis: We Are Likely Overlooking the Impact of Distributional Assumptions, *Ecol. Modell.*, **340**:57–63, 2016.
36. Spearman, C., The Proof and Measurement of Association between Two Things, *Am. J. Psychol.*, **15**(1):72–101, 1904.
37. Kendall, M.G. and Smith, B.B., The Problem of m Rankings, *Ann. Math. Stat.*, **10**(3):275–287, 1939.
38. Savage, I.R., Contributions to the Theory of Rank Order Statistics: The Two Sample Case, *Ann. Math. Stat.*, **27**:590–615, 1956.
39. Razavi, S. and Gupta, H.V., A New Framework for Comprehensive, Robust, and Efficient Global Sensitivity Analysis: 1. Theory, *Water Resour. Res.*, **52**(1):423–439, 2016.

40. Owen, A.B., Better Estimation of Small Sobol' Sensitivity Indices, *ACM Trans. Modell. Comput. Simul.*, **23**(2):1–17, 2013.
41. Iooss, B., Janon, A., Pujol, G., Baptiste, B., Boumhaout, K., Veiga, S.D., Delage, T., Amri, R.E., Fruth, J., Gilquin, L., Guillaume, J., Le Gratiet, L., Lemaitre, P., Marrel, A., Meynaoui, A., Nelson, B.L., Monari, F., Oomen, R., Rakovec, O., Ramos, B., Roustant, O., Song, E., Staum, J., Sueur, R., Touati, T., and Weber, F., Sensitivity: Global Sensitivity Analysis of Model Outputs, R package version 1.22.1, 2020.
42. Puy, A., Lo Piano, S., and Saltelli, A., A Sensitivity Analysis of the PAWN Sensitivity Index, *Env. Modell. Software*, **127**:104679, 2020.
43. Owen, A.B., Randomly Permuted (t, m, s)-Nets and (t, s)-Sequences, in *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, Vol. 106, New York: Springer, pp. 299–317, 1995.
44. Box, G.E.P., Hunter, J.S., and Hunter, W.G., *Statistics for Experimenters: Design, Innovation, and Discovery*, Hoboken, NJ: Wiley, 2005.
45. Pareto, V., *Manuale Di Economia Politica*, Vol. 13, Societa Editrice, 1906.
46. Box, G.E.P. and Meyer, R.D., An Analysis for Unreplicated Fractional Factorials, *Technometrics*, **28**(1):11–18, 1986.
47. Azzini, I., Listorti, G., Mara, T.A., and Rosati, R., Uncertainty and Sensitivity Analysis for Policy Decision Making: An Introductory Guide, Luxembourg, 2020.
48. Liu, H., Chen, W., and Sudjianto, A., Relative Entropy Based Method for Probabilistic Sensitivity Analysis in Engineering Design, *J. Mech. Des. Trans. ASME*, **128**(2):326–336, 2006.
49. Borgonovo, E., A New Uncertainty Importance Measure, *Reliab. Eng. System Safety*, **92**(6):771–784, 2007.
50. Pianosi, F. and Wagener, T., A Simple and Efficient Method for Global Sensitivity Analysis Based on Cumulative Distribution Functions, *Env. Modell. Software*, **67**:1–11, 2015.
51. Pianosi, F. and Wagener, T., Distribution-Based Sensitivity Analysis from a Generic Input-Output Sample, *Env. Modell. Software*, **108**:197–207, 2018.
52. Puy, A., R Code of "A Comprehensive Comparison of Total-Order Estimators for Global Sensitivity Analysis", *Zenodo*, p. 4946559, 2021.
53. Puy, A., Lo Piano, S., Saltelli, A., and Levin, S.A., Sensobol: An R Package to Compute Variance-Based Sensitivity Indices, *Stat. Comput.*, arXiv:2101.10103, 2021.

SUPPLEMENT 1. RAZAVI AND GUPTA ESTIMATOR (VARS)

Unlike the other total-order estimators examined in our paper, the Razavi and Gupta VARS (for variogram analysis of response surfaces [20,39]) relies on the variogram $\gamma(.)$ and covariogram $C(.)$ functions to compute what they call the VARS total-order (VARS-TO) index.

Let us consider a function of factors $\boldsymbol{x} = (x_1, x_2, ..., x_k) \in \mathbb{R}^k$. If $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$ are two generic points separated by a distance $\boldsymbol{h}$, then the variogram is calculated as follows:

$$\gamma(\boldsymbol{x}_A - \boldsymbol{x}_B) = \frac{1}{2} V[y(\boldsymbol{x}_A) - y(\boldsymbol{x}_B)] \tag{S.1}$$

and the covariogram as

$$C(\boldsymbol{x}_A - \boldsymbol{x}_B) = \text{COV}[y(\boldsymbol{x}_A), y(\boldsymbol{x}_B)] \tag{S.2}$$

Note that

$$V[y(\boldsymbol{x}_A) - y(\boldsymbol{x}_B)] = V[y(\boldsymbol{x}_A)] + V[y(\boldsymbol{x}_B)] - 2\text{COV}[y(\boldsymbol{x}_A), y(\boldsymbol{x}_B)] \tag{S.3}$$

and since $V[y(\boldsymbol{x}_A)] = V[y(\boldsymbol{x}_B)]$, then

$$\gamma(\boldsymbol{x}_A - \boldsymbol{x}_B) = V[y(\boldsymbol{x})] - C(\boldsymbol{x}_A, \boldsymbol{x}_B) \tag{S.4}$$

In order to obtain the total-order effect T_i, the variogram and covariogram are computed on all couples of points spaced h_i along the x_i-axis, with all other factors being kept fixed. Thus, Eq. (S.4) becomes

$$\gamma_{x^*_{\sim} i}(h_i) = V(y|x^*_{\sim} i) - C_{x^*_{\sim} i}(h_i) \tag{S.5}$$

where $x^*_{\sim} i$ is a fixed point in the space of non-x_i. [20,39] suggest to take the mean value across the factors' space on both sides of Eq. (S.5), thus obtaining

$$E_{x^*_{\sim} i}\big[\gamma_{x^*_{\sim} i}(h_i)\big] = E_{x^*_{\sim} i}[V(y|x^*_{\sim i})] - E_{x^*_{\sim} i}\big[C_{x^*_{\sim} i}(h_i)\big] \tag{S.6}$$

which can also be written as follows:

$$E_{x^*_{\sim} i}\big[\gamma_{x^*_{\sim} i}(h_i)\big] = V(y)T_i - E_{x^*_{\sim} i}\big[C_{x^*_{\sim} i}(h_i)\big] \tag{S.7}$$

and therefore

$$T_i = \frac{E_{x^*_{\sim} i}\big[\gamma_{x^*_{\sim} i}(h_i)\big] + E_{x^*_{\sim} i}\big[C_{x^*_{\sim} i}(h_i)\big]}{V(y)} \tag{S.8}$$

The sampling scheme for VARS does not rely on $\mathbf{A}, \mathbf{B}, \mathbf{A}_B^{(i)}$... matrices, but on star centers and cross sections. Star centers are N random points sampled across the input space. For each of these stars, k cross sections of points spaced Δh apart are generated, including and passing through the star center. Overall, the computational cost of VARS amounts to $N_t = N[k((1/\Delta h) - 1) + 1]$.

SUPPLEMENT 2. FIGURES

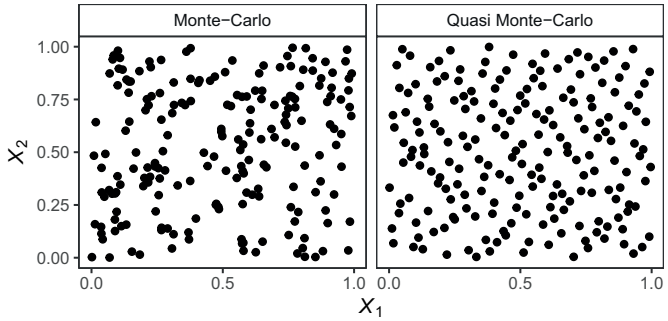

FIG. S1: Examples of Monte Carlo and Quasi Monte Carlo sampling in two dimensions, $N = 2 \times 10^2$

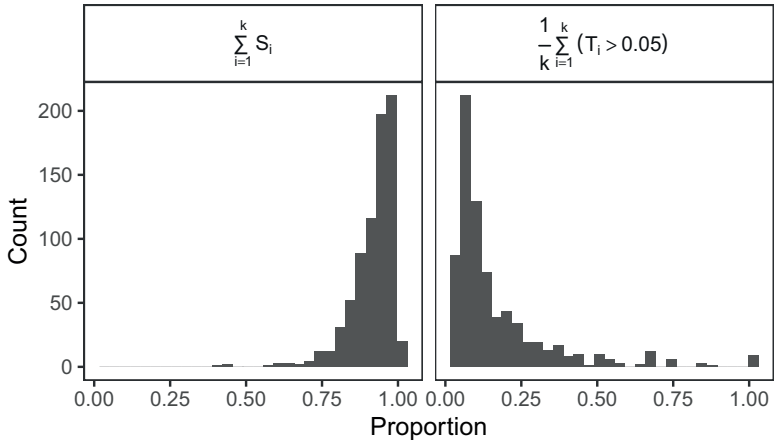

FIG. S2: Proportion of the total sum of first-order effects and of the active model inputs (defined as $T_i > 0.05$) after 10×10^2 random metafunction calls with $k \in (3, 100)$. Note how the sum of first-order effects clusters around 0.8 (thus evidencing the production of non-additivities) and how, on average, the number of active model inputs revolves around 10–20%, thus reproducing the Pareto principle.

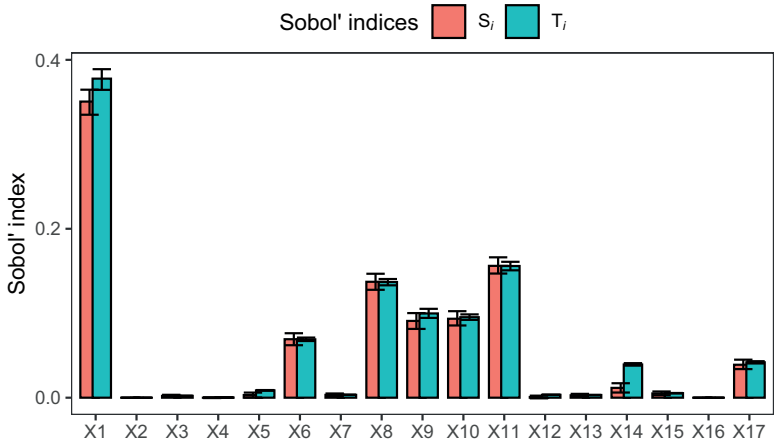

FIG. S3: Sobol' T_i indices obtained after a run of the metafunction with the following parameter settings: $N = 10^4$, $k = 17$, $k_2 = 0.5$, $k_3 = 0.2$, $\varepsilon = 666$. The error bars reflect the 95% confidence intervals after bootstrapping ($R = 10^2$). The indices have been computed with the Jansen [15] estimator.

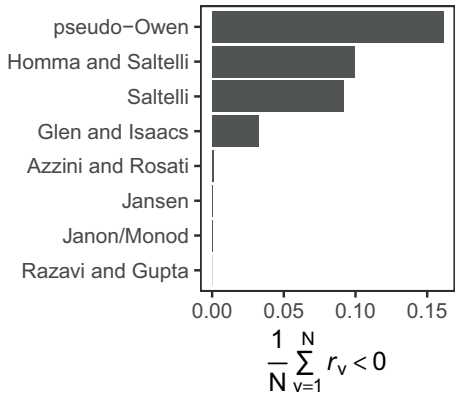

FIG. S4: Proportion of model runs yielding $r < 0$

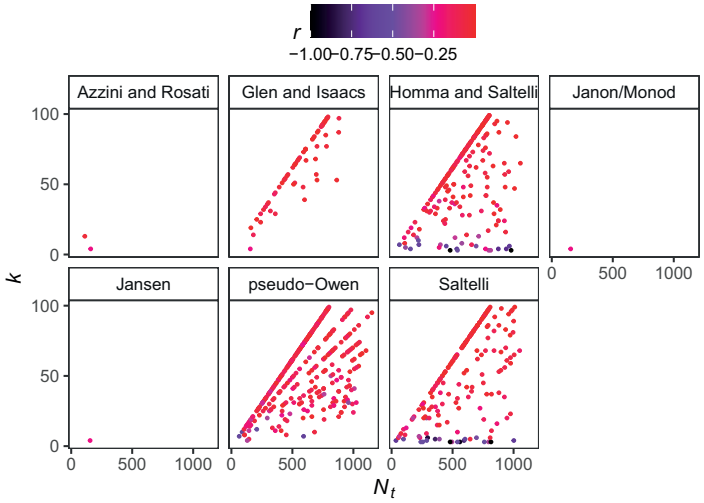

FIG. S5: Scatter of the total number of model runs N_t against the function dimensionality k only for $r < 0$

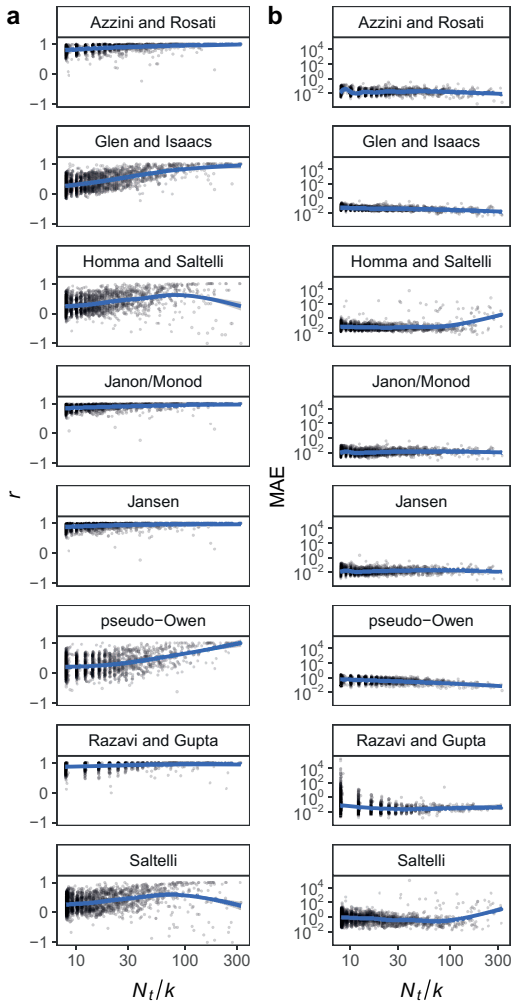

FIG. S6: Scatterplot of the correlation between $\boldsymbol{T}_i$ and $\hat{\boldsymbol{T}}_i$ (r) against the number of model runs allocated per model input (N_t/k)

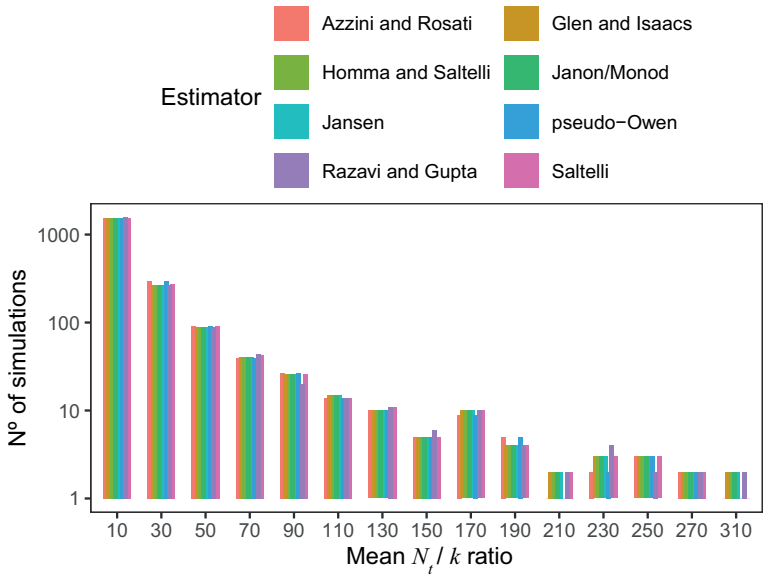

FIG. S7: Bar plot with the number of simulations conducted in each of the N_t/k comparments assessed. All estimators have approximately the same number of simulations in each compartment.

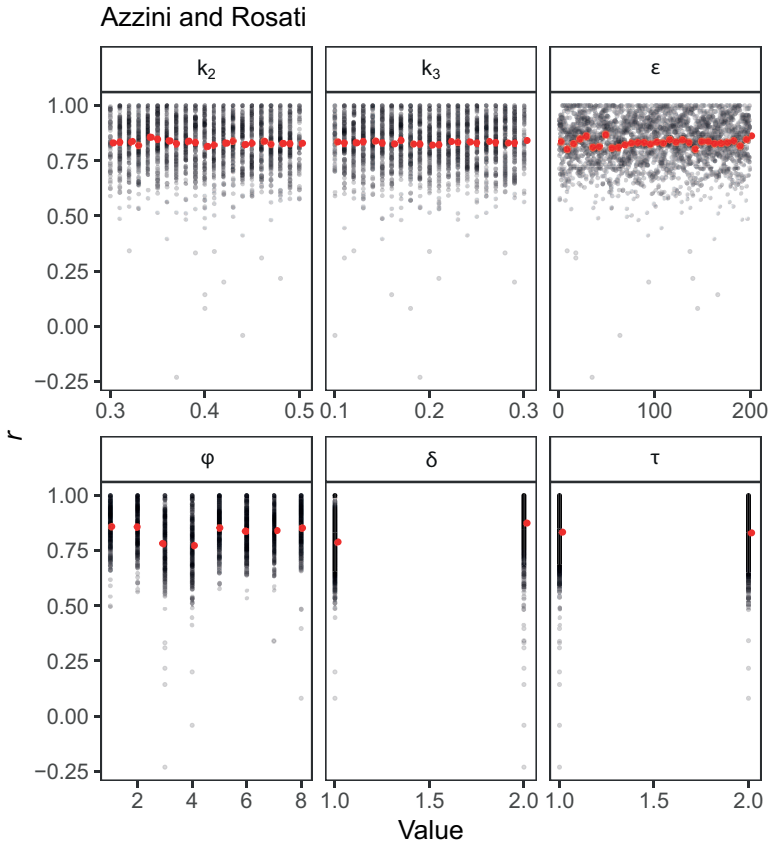

FIG. S8: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

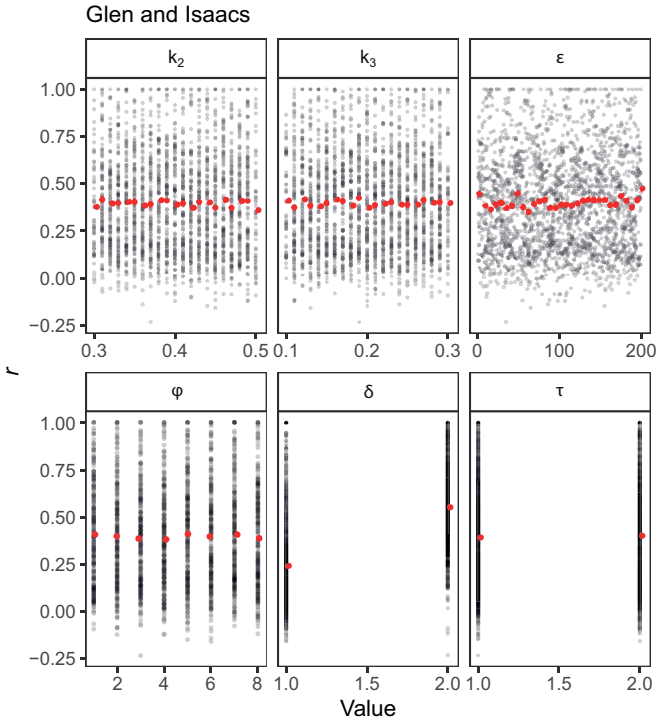

FIG. S9: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

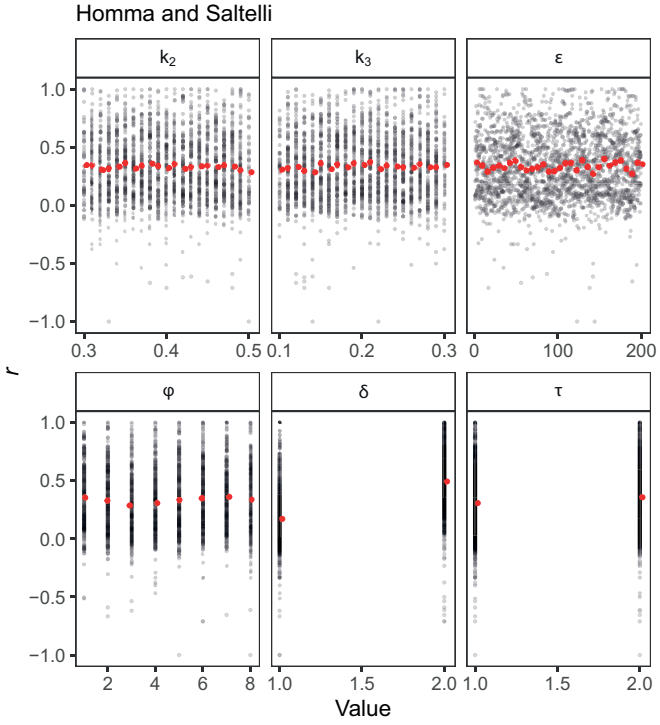

FIG. S10: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

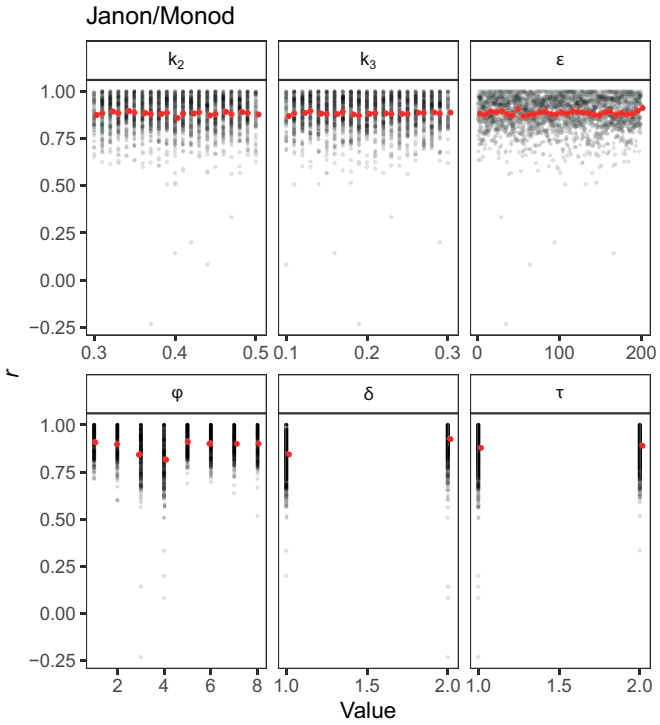

FIG. S11: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

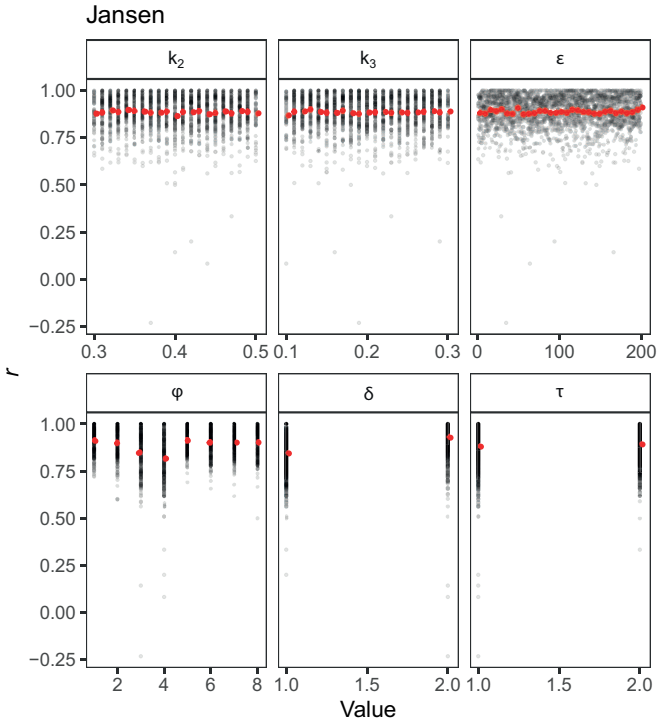

FIG. S12: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

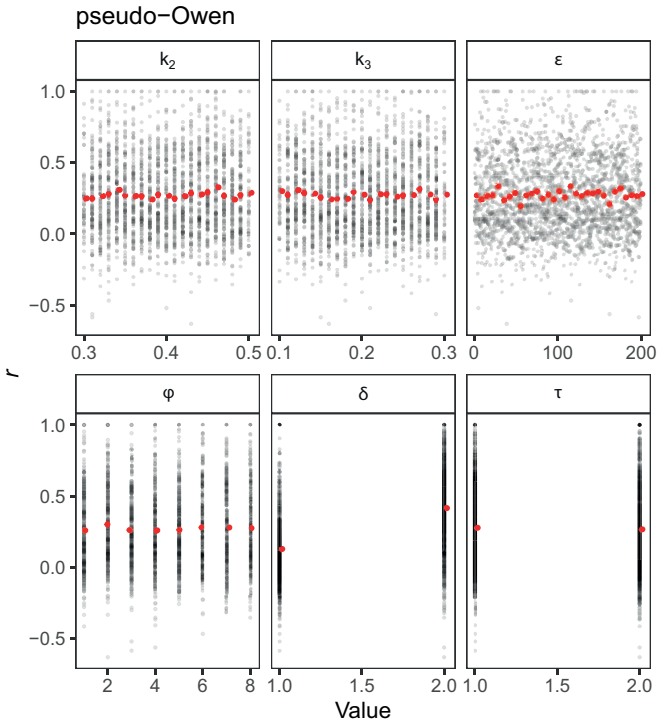

FIG. S13: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

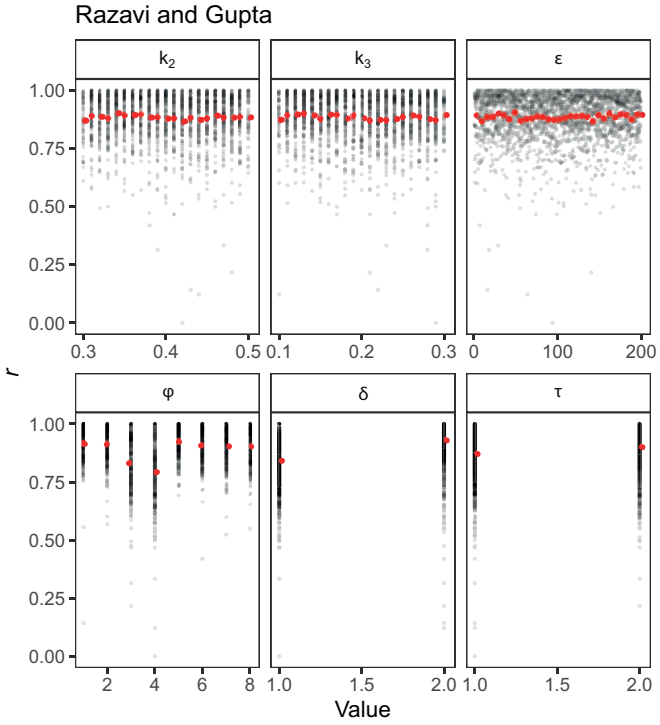

FIG. S14: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

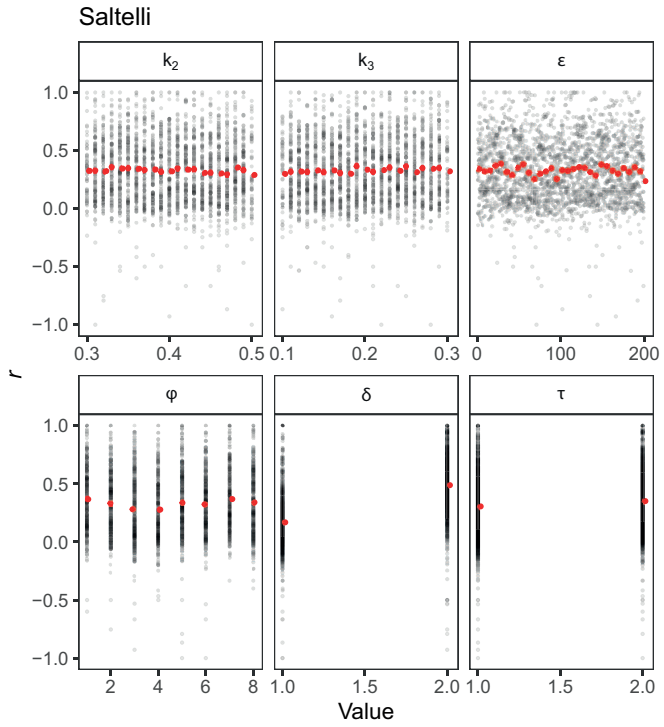

FIG. S15: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

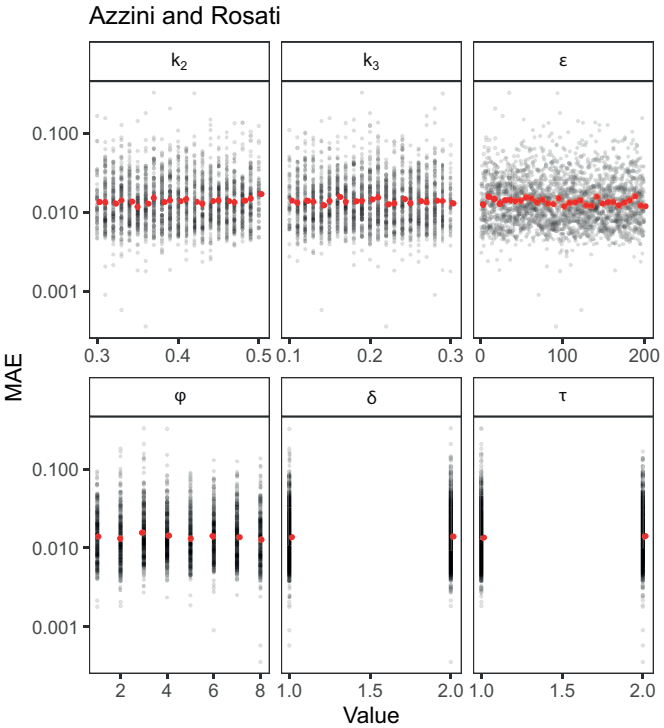

FIG. S16: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

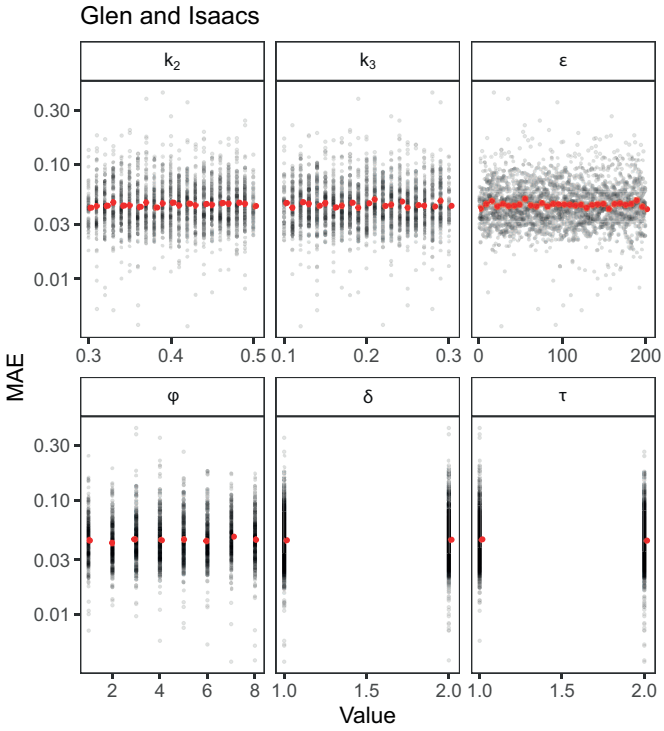

FIG. S17: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

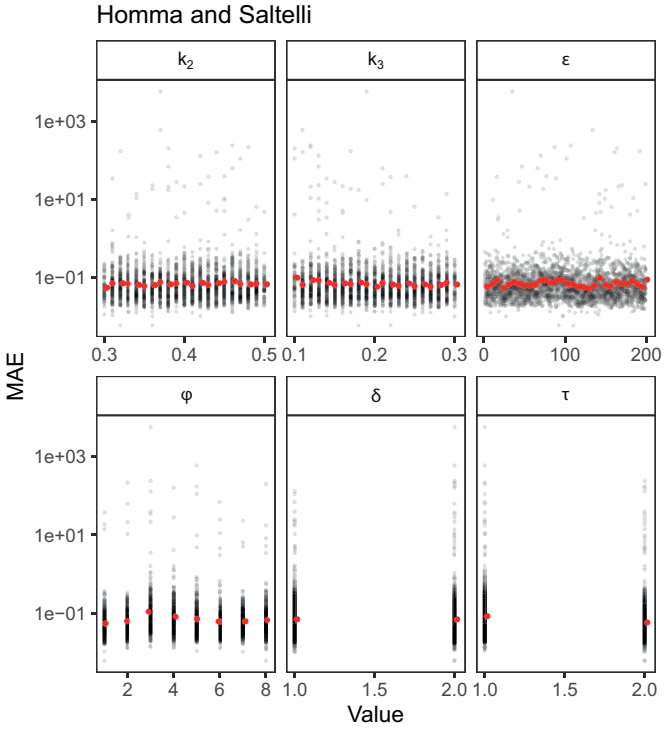

FIG. S18: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

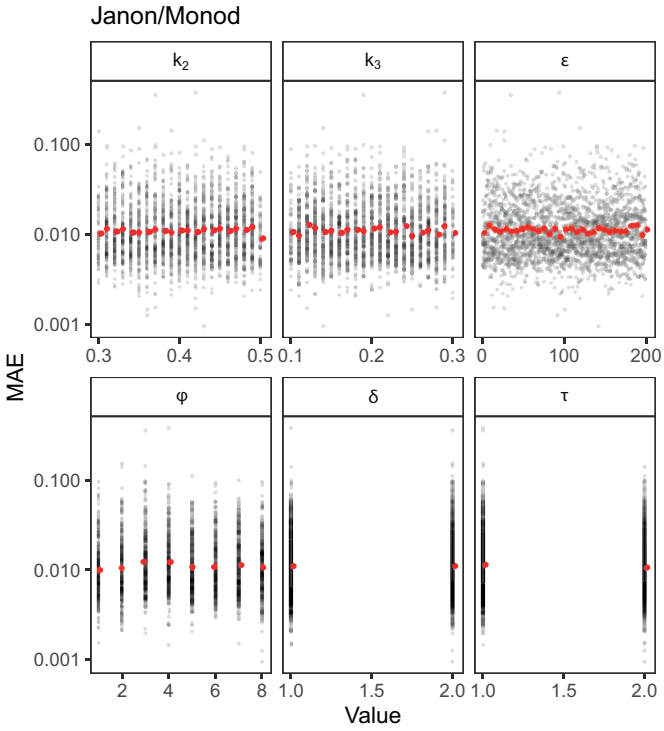

FIG. S19: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

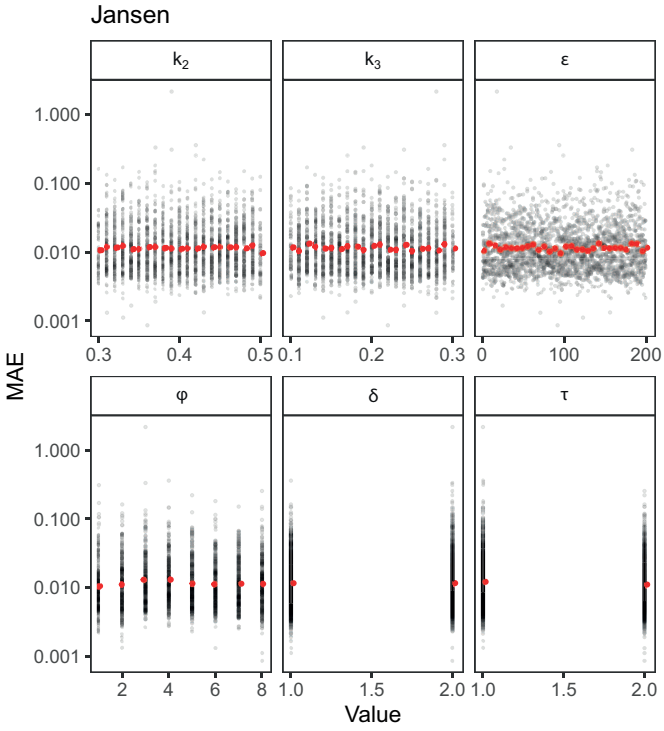

FIG. S20: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

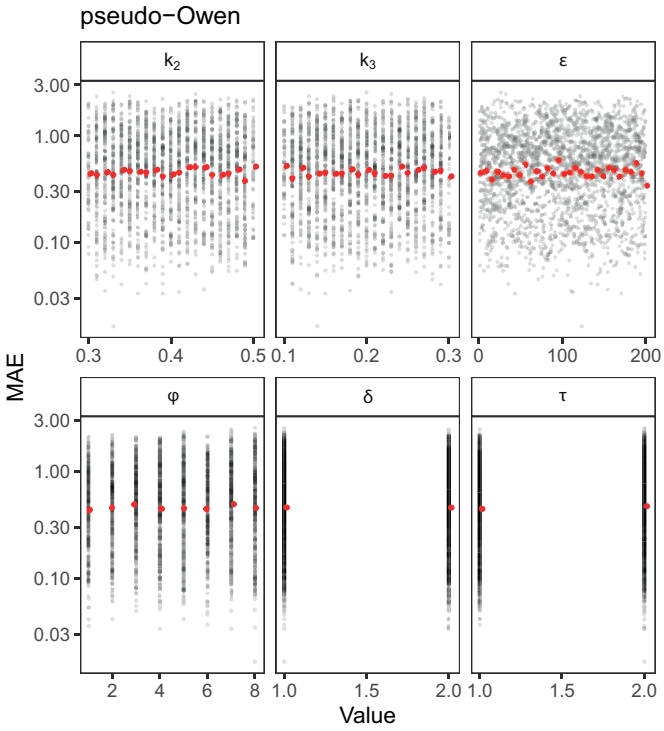

FIG. S21: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

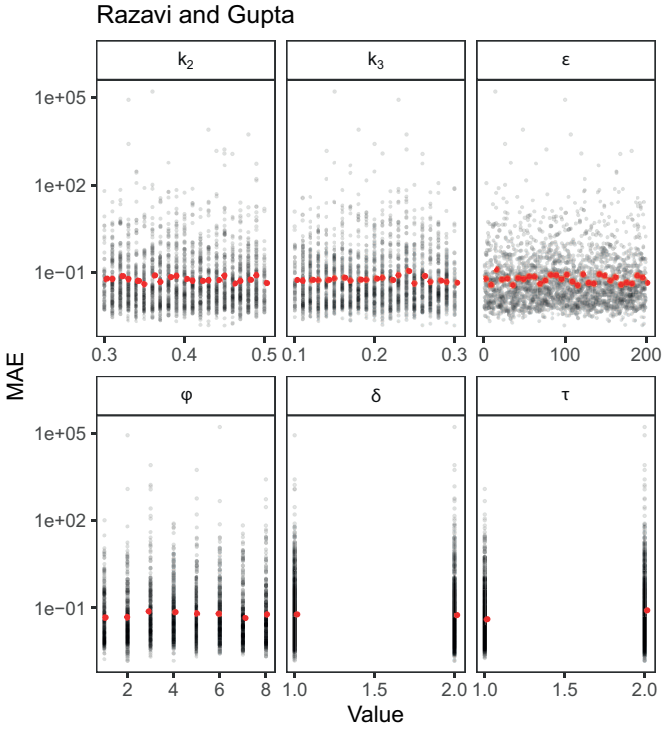

FIG. S22: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).

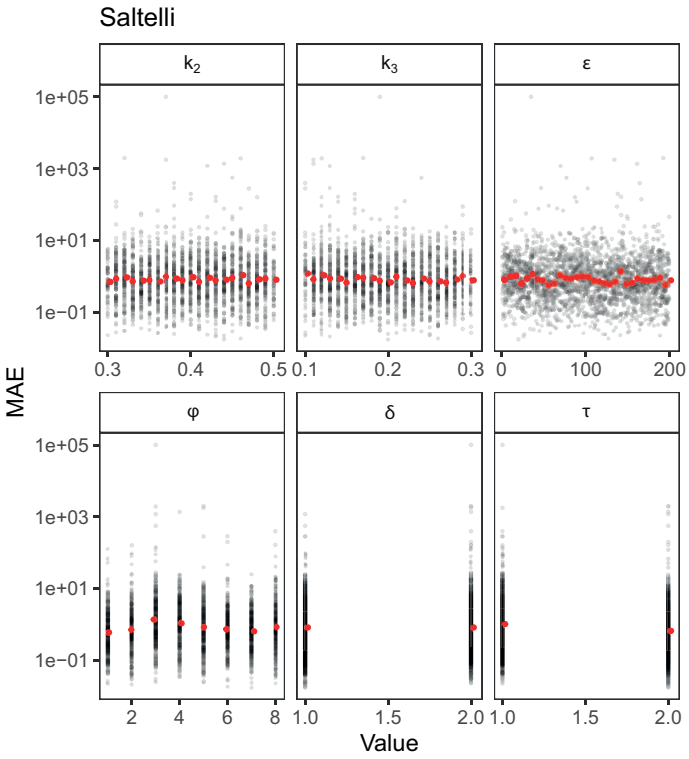

FIG. S23: Scatterplots of the model inputs against the model output. The red dots show the mean value in each bin (we have set the number of bins arbitrarily at 30).