Sensitivity maps of complex systems with a functional output

Mexico Workshop
4-5 Dec 2025

O. Roustant, INSA Toulouse
Joint works with Y.T. Sáo and G. F. Maciel, São Paulo State University

Part I

Introduction

Aim

Let us consider a *time-consuming* simulator.
Assume that the output is a *function* of z = time, space, etc.

Goals

- To **compute *fastly* a sensitivity map (SM)**, which contains the influence of input variables on the output *for all values of* z
- To **account for *estimation* error**
- To **account for *(meta)modeling* error**
  $\rightarrow$ *approximation error of the model, assumed to be well-specified*

A motivating example

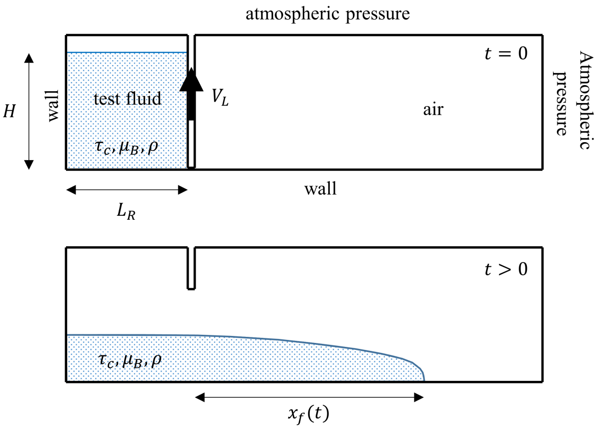


Figure: Schematic of the idealized gradual dam-break case with boundary conditions and input variables.

Sensitivity maps, with estimation error

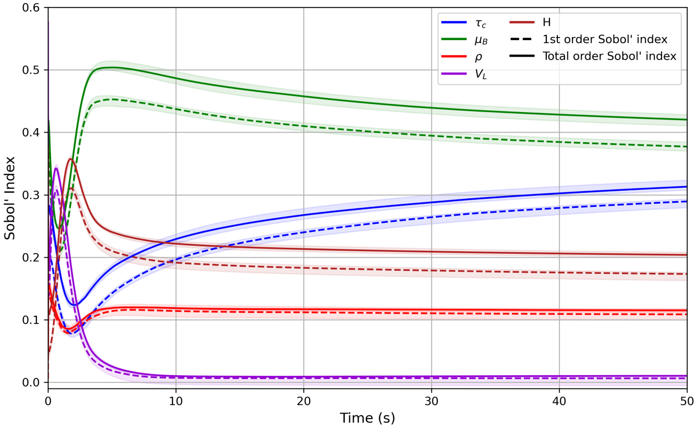


Figure: Total and first-order SMs of all input variables.

Sensitivity maps, with estimation error

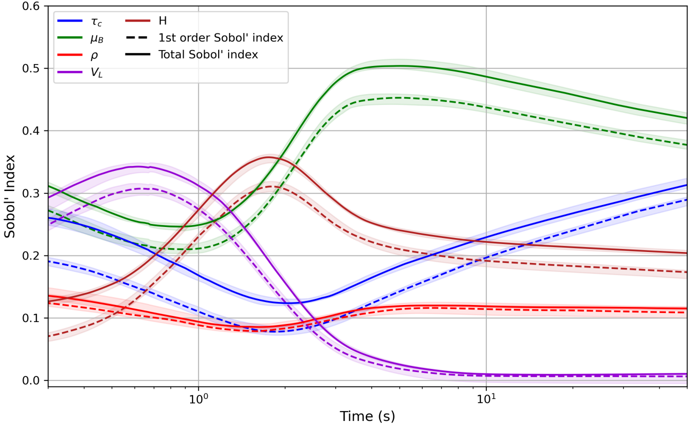


Figure: Total and first-order SMs of all input variables.

Sensitivity maps, with estimation + metamodel error

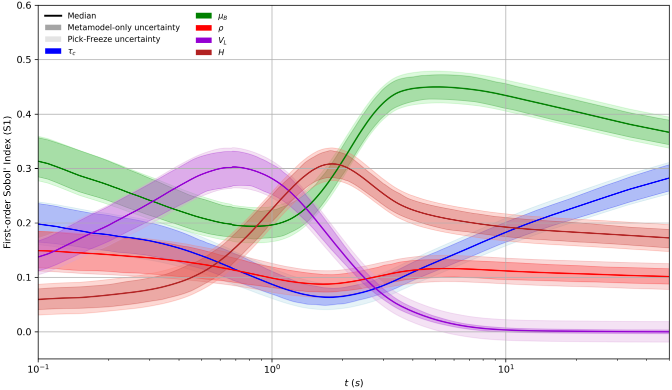


Figure: First-order SMs of all input variables.

Sensitivity maps, with estimation + metamodel error

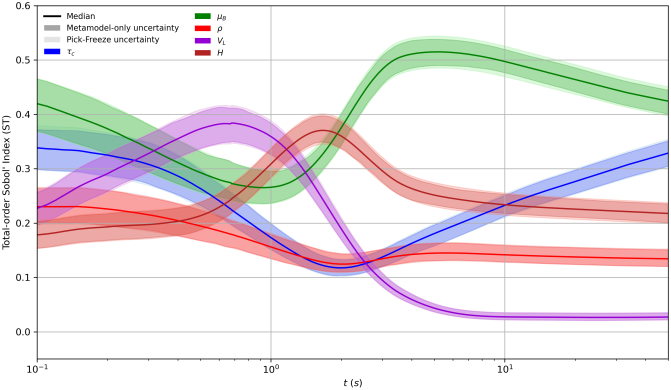


Figure: Total SMs of all input variables.

Part II

Computation of sensitivity maps with basis expansions

Framework and notations

The simulator is modeled by a high-dimensional vector-valued function

$$\begin{array}{ccc} \mathbb{X} \subseteq \mathbb{R}^d & \longrightarrow & \mathbb{R}^L \\ x & \longmapsto & y(x) = (y_\ell(x))_{\ell=1,\dots,L} \end{array} \quad (1)$$

We assume that it is well approximated with a linear basis expansion (e.g. B-splines, wavelet, principal component analysis, etc.)

$$y_\ell(x) \approx \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L \quad (2)$$

The idea is to compute the sensitivity maps of $y(x)$ from the low-dimensional vector $c(x) = (c_1(x), \dots, c_{n_b}(x))$.

Variance-based sensitivity analysis (scalar-valued case)

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a function. Let $X = (X_1, \dots, X_d)$ be a random vector with independent components, and $Y = f(X)$, assumed to be square integrable.

The Sobol-Hoeffding decomposition is written

$$f(X) = \sum_{I \subseteq \{1,\dots,d\}} f_I(X_I)$$

with orthogonal terms, leading to the variance decomposition

$$\mathbb{V}\mathrm{ar}(f(X)) = \sum_{I \subseteq \{1,\dots,d\}} \mathbb{V}\mathrm{ar}(f_I(X_I))$$

Variance-based sensitivity analysis (scalar-valued case)

Denote $D = \mathbb{V}\mathrm{ar}(f(X))$ the overall variance.
We will consider the closed Sobol' indices

$$\underline{D}_I(f) = \mathbb{V}\mathrm{ar}(\mathbb{E}[f(X)|X_I]) \qquad \underline{S}_I(f) = \frac{\underline{D}_I}{D} \tag{3}$$

and the total Sobol' indices

$$\overline{D}_i(f) = \sum_{J \supseteq \{i\}} \mathbb{V}\mathrm{ar}(f_J(X)) \qquad \overline{S}_i(f) = \frac{\overline{D}_i}{D} \tag{4}$$

Example : $I = \{i\}$.

- $\underline{S}_i$ coincides with the first-order Sobol' index, measuring the ratio of variance of $f(X)$ explained only by X_i.
- $\overline{S}_i$ measures the ratio of variance explained by X_i and its interactions. We have $\overline{S}_i = 1 - \underline{S}_{-i}$ where $-i = \{1, \dots, d\} \setminus \{i\}$.

Variance-based sensitivity analysis (vector-valued case)

Let consider a vector-valued function $y : \mathbb{R}^d \to \mathbb{R}^L$. Then all 'variances' are covariance matrices, and the interpretation is more complex.

Thus, the overall covariance and *unnormalized* closed Sobol' index are

$$\mathbf{D}(y) = \mathbb{C}\mathrm{ov}(y(X)) \qquad \underline{\mathbf{D}}_I(y) = \mathbb{C}\mathrm{ov}(\mathbb{E}[y(X)|X_I])$$

A meaningful scalar sensitivity index is the *generalized sensitivity index* (GSI). The closed GSI is defined by

$$\underline{\mathrm{GSI}}_I(y) = \frac{\mathrm{Tr}\left[\mathbb{C}\mathrm{ov}(\mathbb{E}[y(X)|X_I])\right]}{\mathrm{Tr}\left[\mathbb{C}\mathrm{ov}(y(X))\right]} = \frac{\sum_{\ell=1,\dots,L} \underline{D}_I(y_\ell)}{\sum_{\ell=1,\dots,L} D(y_\ell)} \tag{5}$$

Fast computation of sensitivity maps

Assume that the basis expansion is exact:

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

Then,

Fast computation of sensitivity maps

Assume that the basis expansion is exact:

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

Then,

$$\mathbb{E}[y_\ell(X)|X_I] = \sum_{q=1}^{n_b} \mathbb{E}[c_q(X)|X_I] v_{q,\ell}$$

Fast computation of sensitivity maps

Assume that the basis expansion is exact:

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

Then,

$$\begin{aligned}
\mathbb{E}[y_\ell(X)|X_I] &= \sum_{q=1}^{n_b} \mathbb{E}[c_q(X)|X_I] v_{q,\ell} \\
\mathbb{V}\mathrm{ar}(\mathbb{E}[y_\ell(X)|X_I]) &= \sum_{q,q'=1}^{n_b} \underbrace{\mathbb{C}\mathrm{ov}(\mathbb{E}[c_q(X)|X_I], \mathbb{E}[c_{q'}(X)|X_I])}_{\left(\underline{\mathbf{D}}_I(c)\right)_{q,q'}} v_{q,\ell} v_{q',\ell}
\end{aligned}$$

Fast computation of sensitivity maps

Assume that the basis expansion is exact:

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

Then,

$$\begin{aligned}
\mathbb{E}[y_\ell(X)|X_I] &= \sum_{q=1}^{n_b} \mathbb{E}[c_q(X)|X_I] v_{q,\ell} \\
\mathbb{V}\mathrm{ar}(\mathbb{E}[y_\ell(X)|X_I]) &= \sum_{q,q'=1}^{n_b} \underbrace{\mathbb{C}\mathrm{ov}(\mathbb{E}[c_q(X)|X_I], \mathbb{E}[c_{q'}(X)|X_I])}_{\left(\underline{\mathbf{D}}_I(c)\right)_{q,q'}} v_{q,\ell} v_{q',\ell}
\end{aligned}$$

i.e., matricially,

$$\underline{D}_I(y_\ell) = v_{.,\ell}^\top \, \underline{\mathbf{D}}_I(c) \, v_{.,\ell}$$

Fast computation of sensitivity maps

Proposition ([Li et al., 2020, Sáo et al., 2026])

The sensitivity map of $\mathbf{y}(X)$ *can be computed with the basis coefficients as*

$$\underline{S}_I(y_\ell) = \frac{v_{\cdot,\ell}^\top \, \underline{\mathbf{D}}_I(c) \, v_{\cdot,\ell}}{v_{\cdot,\ell}^\top \, \mathbf{D}(c) \, v_{\cdot,\ell}}, \qquad \ell = 1, \dots, L \tag{6}$$

Computing $\underline{\mathbf{D}}_I(c)$ is much cheaper than directly computing $\underline{S}_I(y_\ell)$ for all ℓ.

Fast computation of GSI

Proposition ([Sáo et al., 2026])

Generalized sensitivity indices can be computed from the basis coefficients as

$$\underline{\mathrm{GSI}}_I(y) = \frac{\mathrm{Tr}\,[\underline{\mathbf{D}}_I(c)\,\mathbf{G}]}{\mathrm{Tr}\,[\mathbf{D}(c)\,\mathbf{G}]} \qquad (7)$$

where $\mathbf{G}$ *is the Gram matrix of the basis vectors:* $\mathbf{G}_{q,q'} = \langle \mathbf{v}_{q,.}, \mathbf{v}_{q',.} \rangle_{\mathbb{R}^L}$

A similar expression can be found in [Perrin et al., 2021] for functional PCA.

Part III

Estimation of sensitivity maps with basis expansions

General remark

In general the **estimate** of sensitivity maps may be **computed pixel by pixel** (or time by time) **from the basis expansion** (see e.g. [Marrel et al., 2011])

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

We can do **much better** when using **pick-freeze estimators**.

Pick-freeze estimators (closed Sobol' index)

The pick-freeze (PF) estimators are built on two independent copies X, X' drawn from μ_X, and the evaluations

$$Y = f(X) = f(X_I, X_{-I}), \qquad Y^I = f(X_I, X'_{-I}).$$

Then, we have:

$$\underline{D}_I(Y) = \mathbb{C}\mathrm{ov}(Y, Y^I)$$

Pick-freeze estimators (closed Sobol' index)

Now, consider two indep. N samples, $X^1, \ldots, X^N$ and $X^{*1}, \ldots, X^{*N}$ of μ. Denote

$$Y^k = f\left(X^k\right), \qquad Y^{I,k} = f\left(X_I^k, X_{-I}^{*k}\right) \qquad (k = 1, \ldots, N).$$

Using the empirical version of mean, covariance leads to the PF estimator

$$\widehat{\underline{S}_I}^{\mathrm{pf}} = \frac{\widehat{\underline{D}_I}^{\mathrm{pf}}}{\widehat{D}^{\mathrm{pf}}}$$

with

$$\widehat{\underline{D}_I}^{\mathrm{pf}} = \frac{1}{N}\sum_{k=1}^{N} Y^k Y^{I,k} - \left(\widehat{f_0}^{\mathrm{pf}}\right)^2, \quad \widehat{f_0}^{\mathrm{pf}} = \frac{1}{N}\sum_{k=1}^{N}\left[\frac{Y^k + Y^{I,k}}{2}\right]$$

$$\widehat{D}^{\mathrm{pf}} = \frac{1}{N}\sum_{k=1}^{N}\left[\frac{(Y^k)^2 + (Y^{I,k})^2}{2}\right] - \left(\widehat{f_0}^{\mathrm{pf}}\right)^2$$

This estimator is unbiased, asympt. normal and efficient [Janon et al., 2014, Gamboa et al., 2014]

Pick-freeze estimators (vector-valued case)

The definition of the PF estimator is immediately extended to the vector-valued case, replacing all products of the form $YY^\star$ by $Y(Y^\star)^\top$, e.g.

$$\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}} = \frac{1}{N}\sum_{k=1}^{N} Y^k (Y^{I,k})^\top - \widehat{f_0}^{\mathrm{pf}} \left(\widehat{f_0}^{\mathrm{pf}}\right)^\top$$

Notice that $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}$ is a **quadratic form** of the PF sample

$$Y^1, \dots, Y^N, Y^{I,1}, \dots, Y^{I,N}$$

In particular, with $y_\ell(x) = v_{.,\ell}^\top c(x)$, we obtain

$$Y^k = v_{.,\ell}^\top C^k, \qquad Y^{I,k} = v_{.,\ell}^\top C^{*k}$$

and thus

$$\widehat{\underline{D}_I}^{\mathrm{pf}}(y_\ell) = v_{.,\ell}^\top \widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c) v_{.,\ell}$$

Fast computation of sensitivity maps

Assume that the basis expansion is exact:

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

Proposition ([Sáo et al., 2026])

The PF estimator of the closed sensitivity map can be computed in function of the (matrix-valued) PF estimators of the vector of basis coefficients:

$$\widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell) = \frac{v_{\cdot,\ell}^\top \, \widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c) \, v_{\cdot,\ell}}{v_{\cdot,\ell}^\top \, \widehat{\mathbf{D}}^{\mathrm{pf}}(c) \, v_{\cdot,\ell}}$$

Using $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ is enough to estimate $\underline{S}_I(y_\ell)$ for all ℓ.

Theoretical numerical gain

Method	Computational cost
Dimension-wise approach	$\approx 4n_b \times N \times L$
Basis-derived approach	$\approx 3n_b^2 \times (2N + L)$

The basis-derived approach is faster by a factor of (at least)

$$\frac{H(2N, L)}{3n_b}$$

where $H(.,.)$ is the harmonic mean, $H(a, b) = \left(\frac{a^{-1}+b^{-1}}{2}\right)^{-1}$.

The smaller n_b is compared to N, L, the higher the gain.

Assessment of the estimation error

- Obtain by **bootstrap** a B-sample of the law of $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ and $\widehat{\mathbf{D}}^{\mathrm{pf}}(c)$ by repeating B times:
  - Draw $k_1, \dots, k_N$ independently from $\mathcal{U}\{1, \dots, n\}$
  - Compute $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ and $\widehat{\mathbf{D}}^{\mathrm{pf}}(c)$ replacing the PF sample $Y^1, Y^{I,1}, \dots, Y^N, Y^{I,N}$ by the resampled one $Y^{k_1}, Y^{I,k_1}, \dots, Y^{k_N}, Y^{I,k_N}$

Assessment of the estimation error

- Obtain by **bootstrap** a B-sample of the law of $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ and $\widehat{\mathbf{D}}^{\mathrm{pf}}(c)$ by repeating B times:
  - Draw $k_1, \dots, k_N$ independently from $\mathcal{U}\{1, \dots, n\}$
  - Compute $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ and $\widehat{\mathbf{D}}^{\mathrm{pf}}(c)$ replacing the PF sample $Y^1, Y^{I,1}, \dots, Y^N, Y^{I,N}$ by the resampled one $Y^{k_1}, Y^{I,k_1}, \dots, Y^{k_N}, Y^{I,k_N}$
- Deduce a B-sample of the **joint** law of the vector $(\widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell))_{\ell=1,\dots,L}$ with

$$\widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell) = \frac{v_{.,\ell}^\top \widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)\, v_{.,\ell}}{v_{.,\ell}^\top \widehat{\mathbf{D}}^{\mathrm{pf}}(c)\, v_{.,\ell}}$$

Assessment of the estimation error

- Obtain by **bootstrap** a B-sample of the law of $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ and $\widehat{\mathbf{D}}^{\mathrm{pf}}(c)$ by repeating B times:
  - Draw $k_1, \dots, k_N$ independently from $\mathcal{U}\{1, \dots, n\}$
  - Compute $\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)$ and $\widehat{\mathbf{D}}^{\mathrm{pf}}(c)$ replacing the PF sample $Y^1, Y^{I,1}, \dots, Y^N, Y^{I,N}$ by the resampled one $Y^{k_1}, Y^{I,k_1}, \dots, Y^{k_N}, Y^{I,k_N}$
- Deduce a B-sample of the **joint** law of the vector $(\widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell))_{\ell=1,\dots,L}$ with

$$\widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell) = \frac{v_{.,\ell}^\top \widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}(c)\, v_{.,\ell}}{v_{.,\ell}^\top \widehat{\mathbf{D}}^{\mathrm{pf}}(c)\, v_{.,\ell}}$$

- Main usage: Draw a boxplot of the marginal distributions $\widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell)$

As a by-product, we obtain (at a low computational cost) a B-sample of the law of *statistics involving the joint distribution*, such as $\max_\ell \widehat{\underline{S}_I}^{\mathrm{pf}}(y_\ell)$

Comments

The link between the PF estimators of y_ℓ and c can be extended to *any PF estimator that is a* ***quadratic form*** *of the PF sample*

$$Y^1, Y^{I,1}, \dots, Y^N, Y^{I,N}$$

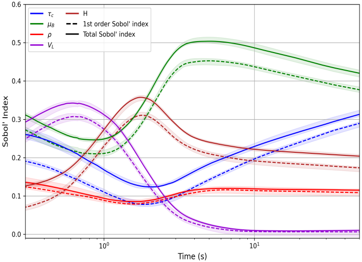
This includes the PF estimator of the total effect from Jansen's formula

$$\widehat{\overline{S}_I}^{\mathrm{pf}} = \frac{\widehat{\overline{D}_I}^{\mathrm{pf}}}{\widehat{D}^{\mathrm{pf}}}, \qquad \widehat{\overline{D}_I}^{\mathrm{pf}} = \frac{1}{2N}\sum_{k=1}^{N}\left(Y^k - Y^{-I,k}\right)^2$$

and the PF estimators of GSI

$$\widehat{\underline{\mathrm{GSI}}_I}^{\mathrm{pf}} := \frac{\mathrm{Tr}\left[\widehat{\underline{\mathbf{D}}_I}^{\mathrm{pf}}\right]}{\mathrm{Tr}\left[\widehat{\mathbf{D}}^{\mathrm{pf}}\right]}, \qquad \widehat{\overline{\mathrm{GSI}}_I}^{\mathrm{pf}} := \frac{\mathrm{Tr}\left[\widehat{\overline{\mathbf{D}}_I}^{\mathrm{pf}}\right]}{\mathrm{Tr}\left[\widehat{\mathbf{D}}^{\mathrm{pf}}\right]}$$

Application to the motivating example



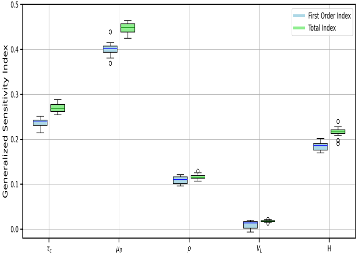


Figure: Sensitivity maps and GSI (time period: [0, 50 s]).

Application to the motivating example

Modeling and computational details

- Modeling
  - We used a space-filling design of experiment of size $\mathbf{n = 200}$
  - The basis expansion used is PCA, with $\mathbf{n_b = 10}$ (**% of variance > 99.9**).
  - Each basis coefficient is modeled by a Gaussian process, independently.
  - The accuracy of the whole model is evaluated on a test set, $Q^2 > 0.9$.
- Sensitivity analysis
  - Input variables are assumed independent and uniformly distributed.
  - Size of pick-freeze samples: $N = 5\,000$.
  - Number of output dimensions: $\mathbf{L = 5\,000}$.
  - Number of bootstrap replicates: 50.
  - Theoretical computational gain : **252**. Observed gain: **35**.
    $\rightarrow$ *Improve coding should even increase the observed gain.*

An example with a spatial output

We consider the Campbell2D function (see e.g. [Marrel et al., 2011]), with:

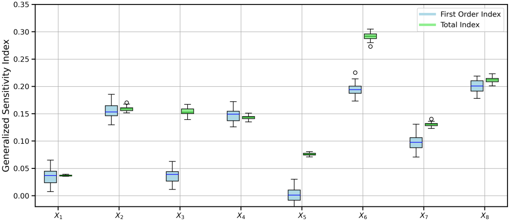
$$f : [-1, 5]^8 \to \mathbb{L}^2([-90, 90]^2)$$

Here we discretize the output on a fine grid of size 64×64, thus $\mathbf{L} = \mathbf{4\,096}$.

The computational setting is similar to the previous application, with the following modifications:

- Modeling
  - For PCA, we chose $\mathbf{n_b} = \mathbf{7}$ (% of variance > 99.2).
  - Here, $Q^2 > 0.95$.
- Sensitivity analysis
  - Theoretical computational gain : **330**. Observed gain: **30**.

An example with a spatial output



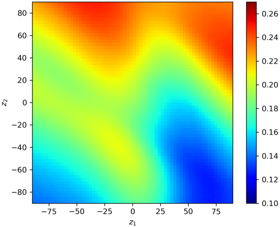


Figure: Left: Estimated GSI - Right: Estimated map of total index for X_8.

Part IV

Estimation of sensitivity maps with model error

Context

In the basis expansion

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

the basis coefficients $c_1(x), \dots, c_{n_b}(x)$ **are only known when x belongs to the design of experiments $\mathcal{X}$**, and must be **predicted elsewhere**.

The model error aims at quantifying the approximation error of $c_1(x), \dots, c_{n_b}(x)$ when $x \notin \mathcal{X}$

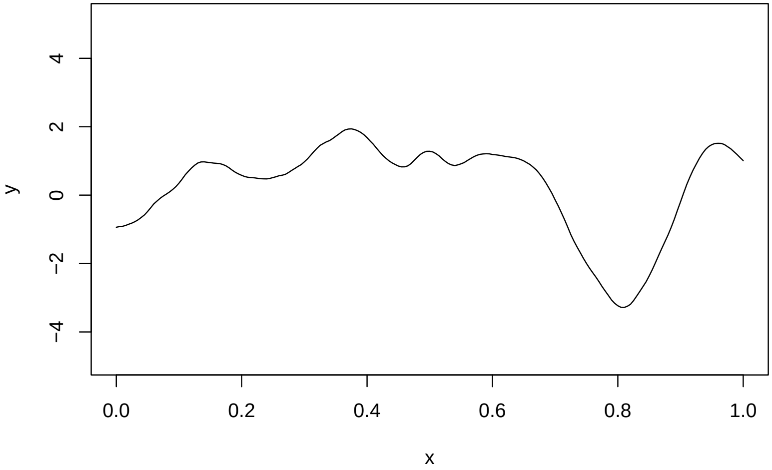
The Gaussian process framework

Let us first consider the scalar output c_q.

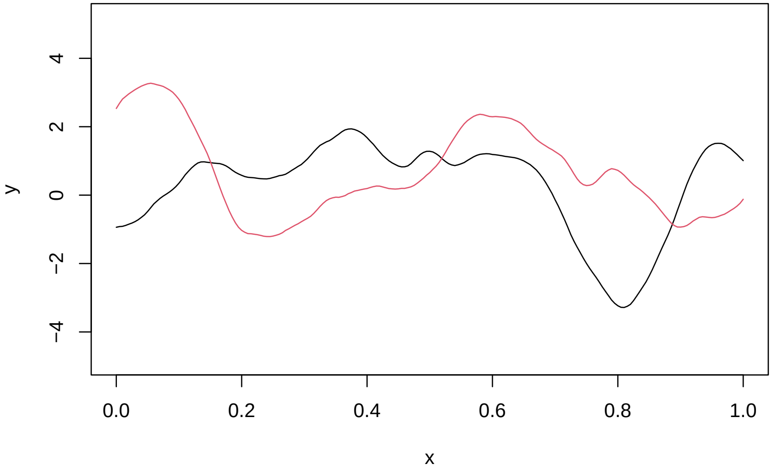
We assume that $x \mapsto c_q(x)$ is a sample path of a Gaussian process (GP) Z.

- (Gaussian prior) $\forall x_1, \dots, x_m$ the law of $(Z(x_1), \dots, Z(x_m))$ is Gaussian. A GP is characterized by
  - a mean function $x \mapsto \mathbb{E}_{\mathbb{P}_{\mathbf{Z}}}(Z(x))$
  - a covariance function (kernel) $k : (x, x') \mapsto \mathbb{C}\text{ov}_{\mathbb{P}_{\mathbf{Z}}}(Z(x), Z(x'))$.
- (Gaussian posterior) $\forall x_1, \dots, x_m$ the law of $(Z(x_1), \dots, Z(x_m))$ conditionally on the data $Z(x^i) = z_i$ $(i = 1, \dots, n)$ is still Gaussian
  - Explicit formulas for the conditional mean and conditional kernel

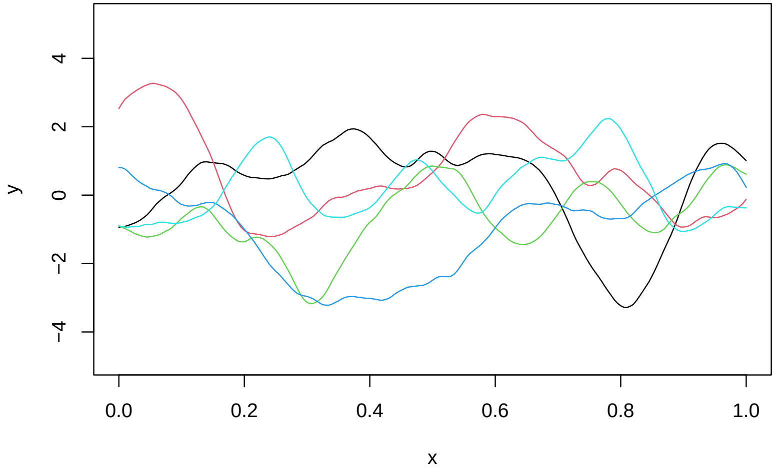
GP simulations



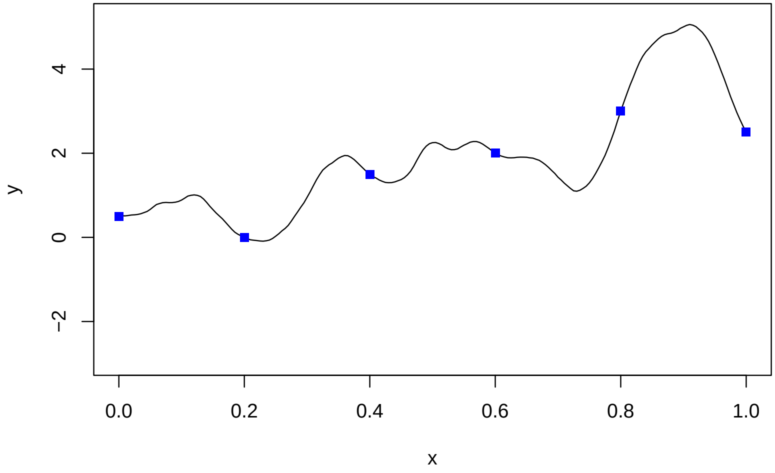
GP simulations



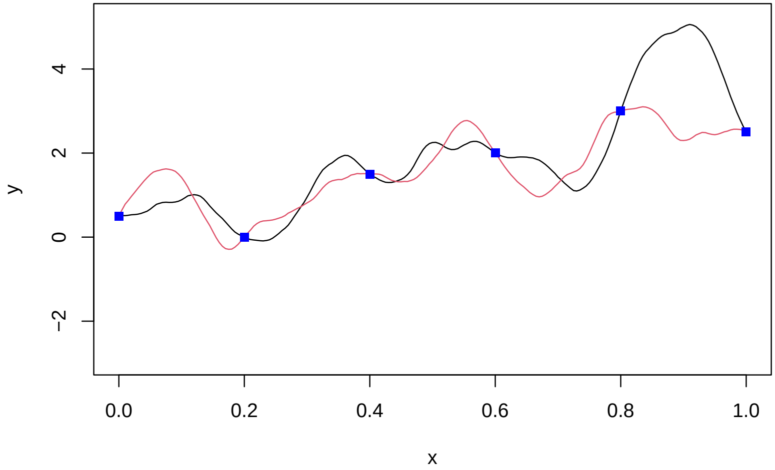
GP simulations



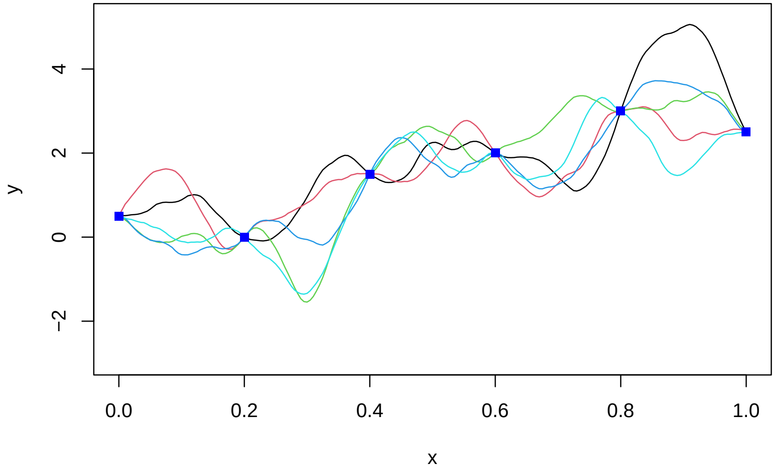
GP conditional simulations



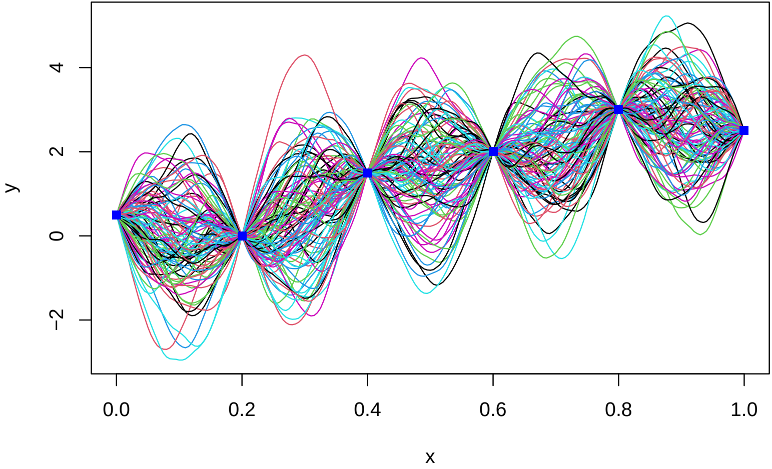
GP conditional simulations



GP conditional simulations



GP conditional simulations



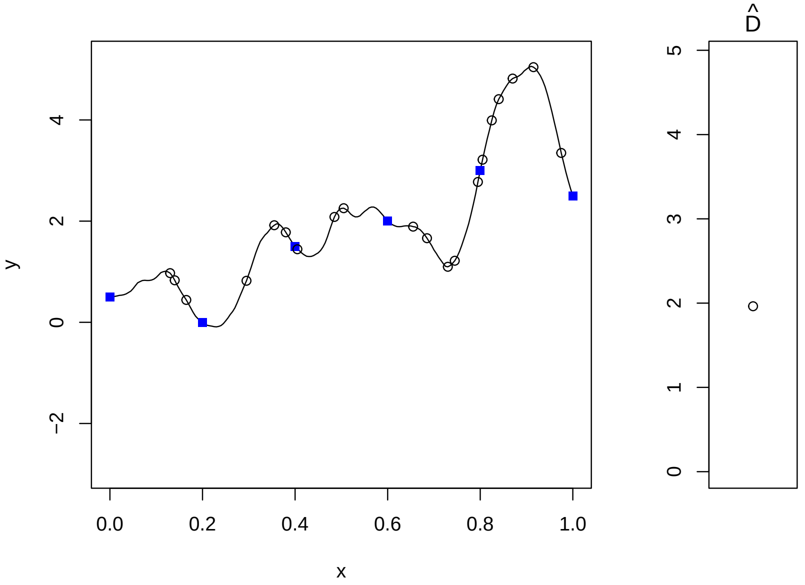
Principle for a scalar output

We now present the methodology proposed by [Le Gratiet et al., 2014], for a scalar output.

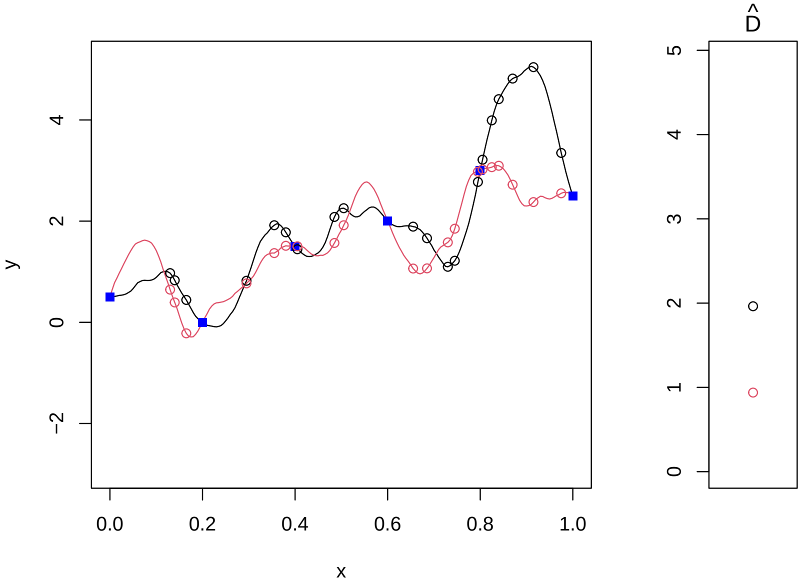
- To assess the **model error**, draw GP sample paths (w.r.t $\mathbb{P}_Z$)
- To assess the **estimation error**, draw PF input samples (w.r.t. $\mathbb{P}_X$) on one GP path
  - $\rightarrow$ Use resampling (bootstrap) to reduce GP sampling cost
- To assess both **model + estimation errors**, draw PF input samples (w.r.t. $\mathbb{P}_X$) on sampled GP paths (w.r.t. $\mathbb{P}_Z$)

y_ℓ.

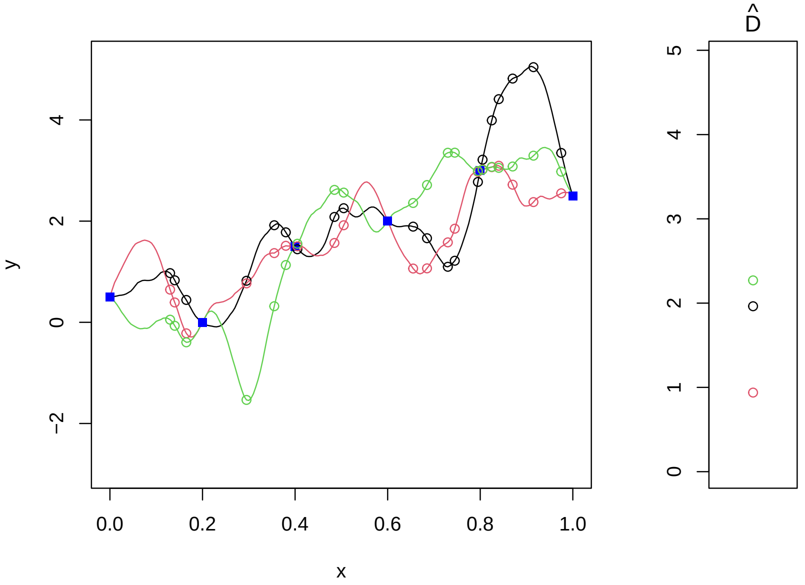
Assessment of model error



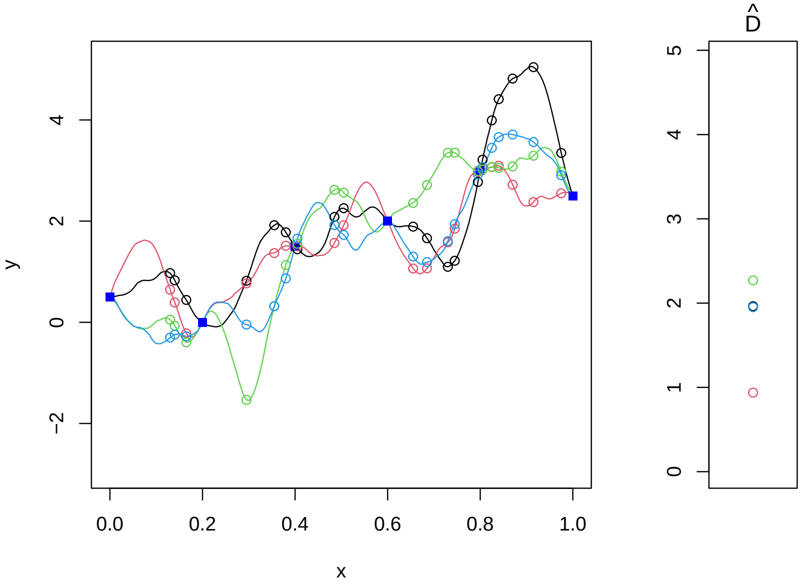
Assessment of model error



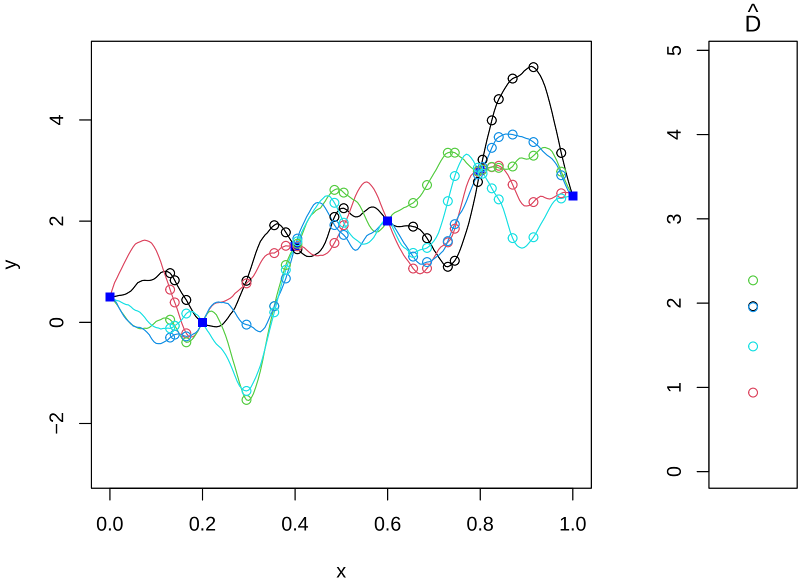
Assessment of model error



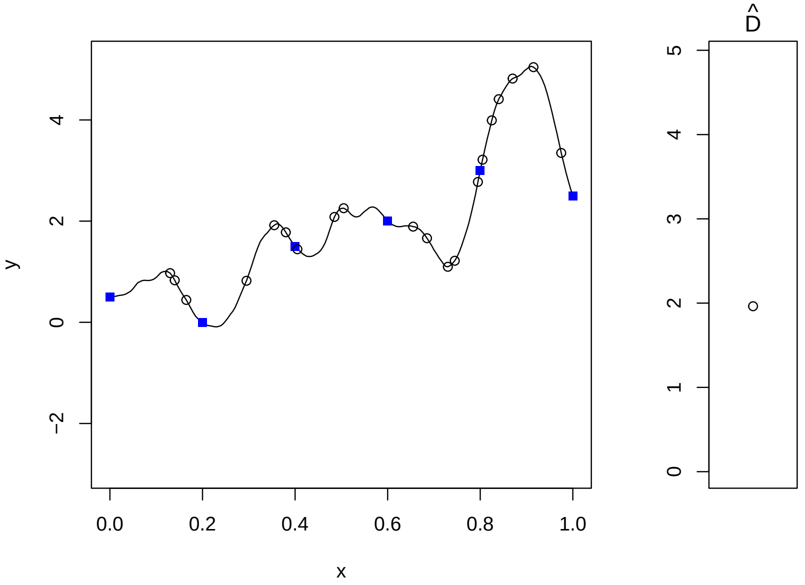
Assessment of model error



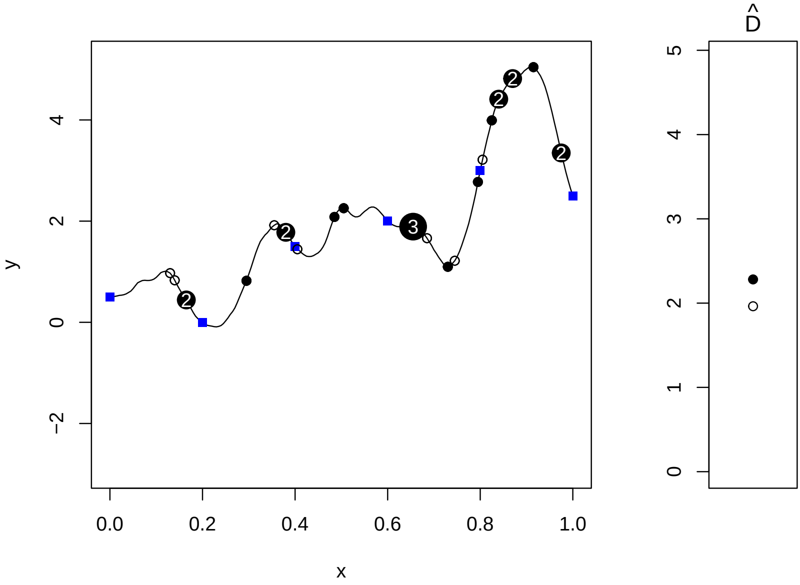
Assessment of model error



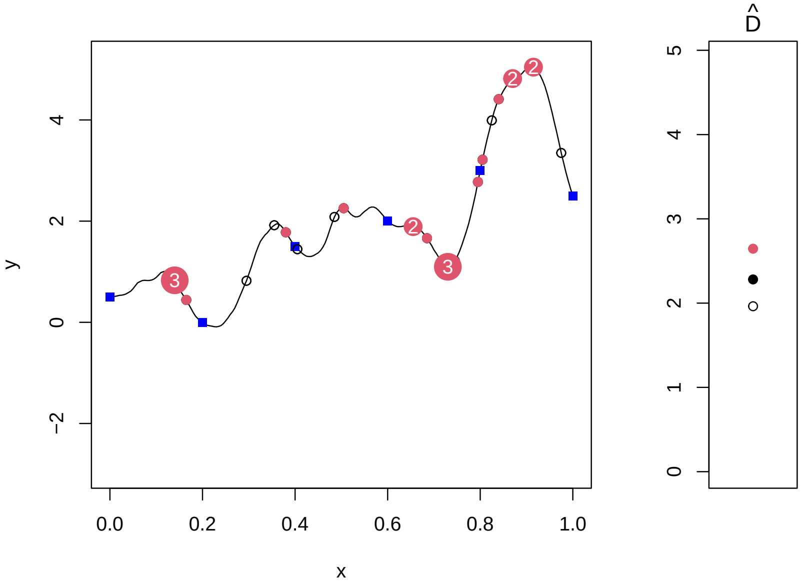
Assessment of estimation error by bootstrap



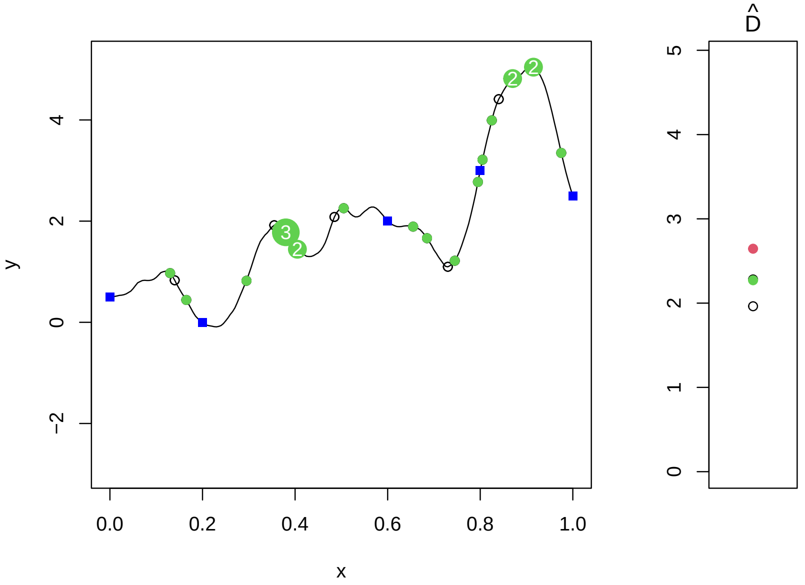
Assessment of estimation error by bootstrap



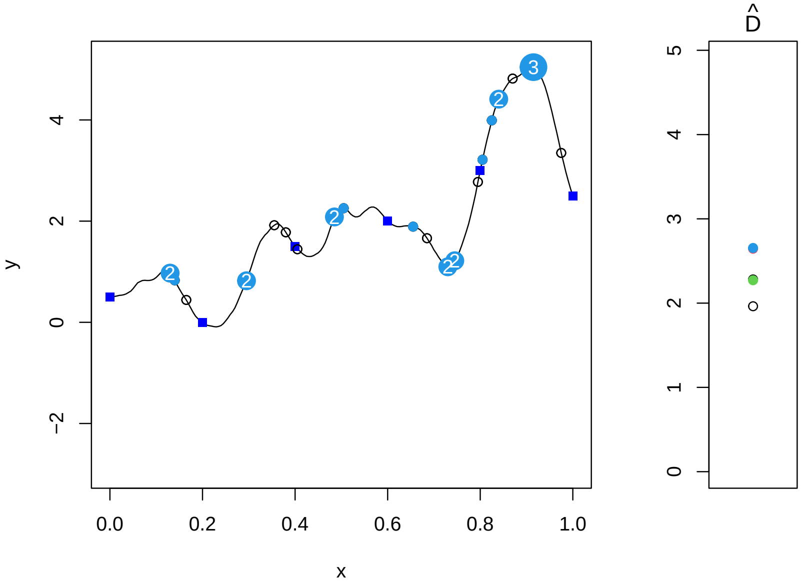
Assessment of estimation error by bootstrap



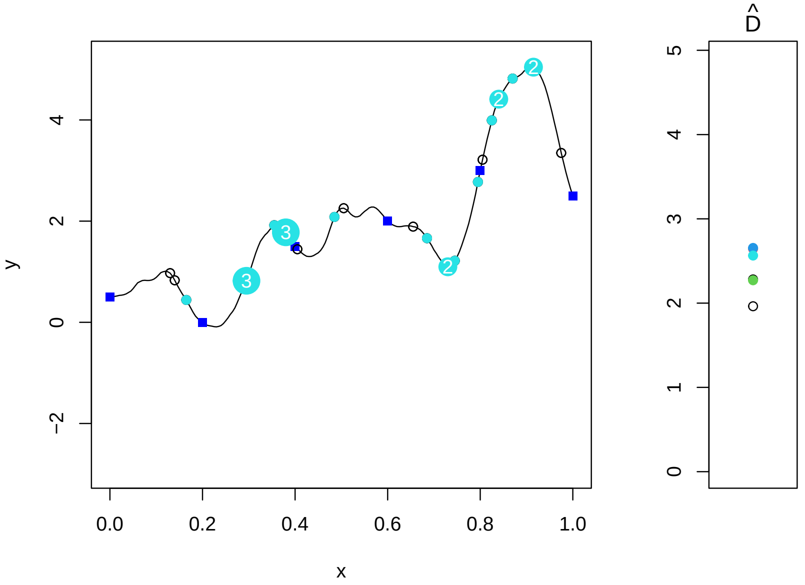
Assessment of estimation error by bootstrap



Assessment of estimation error by bootstrap



Assessment of estimation error by bootstrap



Extension to functional output

We consider the basis expansion:

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

We assume that $x \mapsto c_q(x)$ are sample paths of independent Gaussian processes (for $q = 1, \dots, n_b$).

We can adapt the methodology of [Le Gratiet et al., 2014] with our **fast formulas based on coefficients** to assess model + estimation error on

- sensitivity maps
- GSI

by propagating the errors on the coefficients to all the outputs y_ℓ.

Sensitivity maps, with estimation + metamodel error

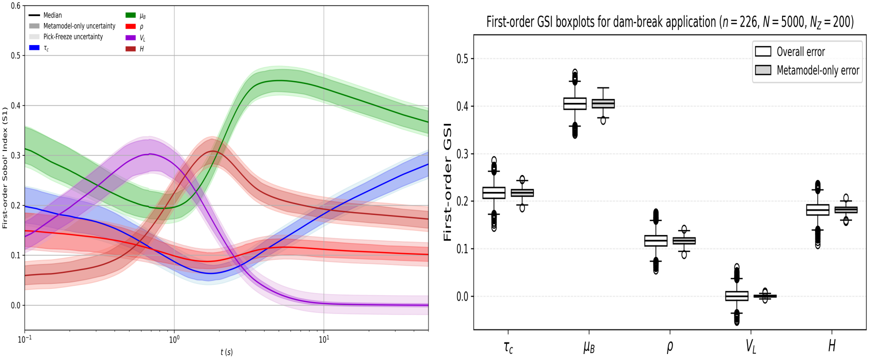
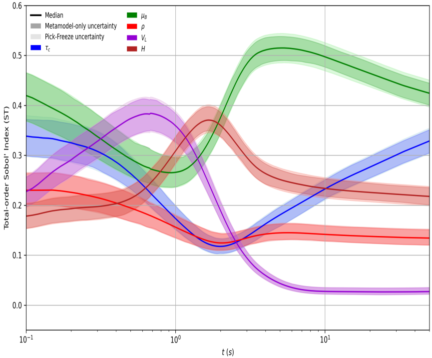


Figure: Estimation of first-order and total SMs, including model error.

Sensitivity maps, with estimation + metamodel error



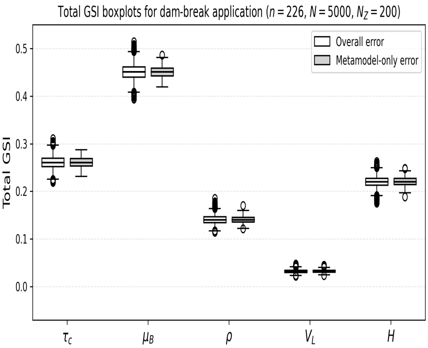


Figure: Estimation of first-order and total SMs, including model error.

Sensitivity maps, with estimation + metamodel error

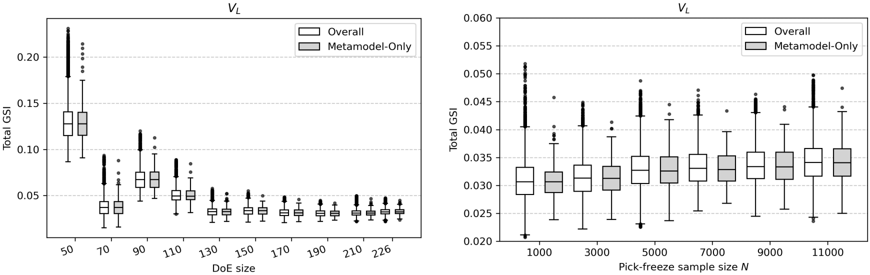


Figure: Total GSI of V_L vs DoE size (N = 5000)/ PF sample size ($n = 226$).

Computational time

The code has been run in an Apple M3 with 8 cores.

- Config 'low': $N = 1\,000$, $N_Z = 10$, $N_X = 10$
- Config 'high': $N = 11\,000$ ($5\,000$ for Campbell), $N_Z = 200$, $N_X = 50$
- S (Standard): Assess the model error for all scalar output dimensions.
- B (Basis-derived): The proposed algorithm, i.e., assess the model error on the vector of basis components and propagate it to the outputs.

\ Method Config \	S	B
low	35s	2 s
high	7h ?	25 mn

Table: Campbell function

\ Method Config \	S	B
low	30s	2 s
high	8h ?	32 mn

Table: Dam-break

Part V

Conclusion and perspectives

Main messages

- Using **any *linear* basis expansion** we obtain **fast formulas of pick-freeze estimates** of (gen.) Sobol' indices for functional outputs
  - → Fast because the PF estimator of the sensitivity map is **deduced from** the PF estimator of **basis coefficients**.
  - → Works for all estimator which is a ***quadratic form* of the PF input samples**
  - → Works for 1st order, 2nd and higher order interactions + total indices.
  - → Gives the **joint distribution of errors**

Main messages

- Using **any *linear* basis expansion** we obtain **fast formulas of pick-freeze estimates** of (gen.) Sobol' indices for functional outputs
  - → Fast because the PF estimator of the sensitivity map is **deduced from** the PF estimator of **basis coefficients**.
  - → Works for all estimator which is a ***quadratic form* of the PF input samples**
  - → Works for 1st order, 2nd and higher order interactions + total indices.
  - → Gives the **joint distribution of errors**
- Using a GP model for basis coefficients, we can also assess **model error**

Main messages

- Using **any *linear* basis expansion** we obtain **fast formulas of pick-freeze estimates** of (gen.) Sobol' indices for functional outputs
  - $\rightarrow$ Fast because the PF estimator of the sensitivity map is **deduced from** the PF estimator of **basis coefficients**.
  - $\rightarrow$ Works for all estimator which is a ***quadratic form* of the PF input samples**
  - $\rightarrow$ Works for 1st order, 2nd and higher order interactions + total indices.
  - $\rightarrow$ Gives the **joint distribution of errors**
- Using a GP model for basis coefficients, we can also assess **model error**
- With other estimators, the estimate of the sensitivity maps may be computed pixel by pixel (or time by time) from the basis expansion

$$y_\ell(x) = \sum_{q=1}^{n_b} c_q(x) v_{q,\ell}, \qquad \ell = 1, \dots, L$$

This is always possible but slower.

A short list of references...

Gamboa, F., Janon, A., Klein, T., and Lagnoux, A. (2014).
Sensitivity analysis for multidimensional and functional outputs.
Electronic Journal of Statistics, 8(1):575 – 603.

Janon, A., Klein, T., Lagnoux, A., Nodet, M., and Prieur, C. (2014).
Asymptotic normality and efficiency of two Sobol' index estimators.
ESAIM: Probability and Statistics, 18:342–364.

Le Gratiet, L., Cannamela, C., and Iooss, B. (2014).
A bayesian approach for global sensitivity analysis of (multifidelity) computer codes.
SIAM/ASA Journal on Uncertainty Quantification, 2(1):336–363.

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

Marrel, A., Iooss, B., Jullien, M., Laurent, B., and Volkova, E. (2011).
Global sensitivity analysis for models with spatially dependent outputs.
Environmetrics, 22(3):383–397.

Perrin, T., Roustant, O., Rohmer, J., Alata, O., Naulin, J., Idier, D., Pedreros, R., Moncoulon, D., and Tinard, P. (2021).
Functional principal component analysis for global sensitivity analysis of model with spatial output.
Reliability Engineering & System Safety, 211:107522.

Sáo, Y. T., Roustant, O., and Maciel, G. d. F. (2026).
Fast pick-freeze estimation of Sobol' sensitivity maps using basis expansions.
Reliability Engineering & System Safety, 265(Part A):111504.

... to be completed by all the references cited in them and many others :)