Statistical methods for the study of computer experiments failures: Application to a fuel-coolant interaction simulation code

F. Hakimi[1,2], C. Brayer[2], A. Marrel[1,3], F. Gamboa[1] , B.Habert[2]

e-mail : faouzi.hakimi@cea.fr

[1]UT3, Université Paul Sabatier, Toulouse, France

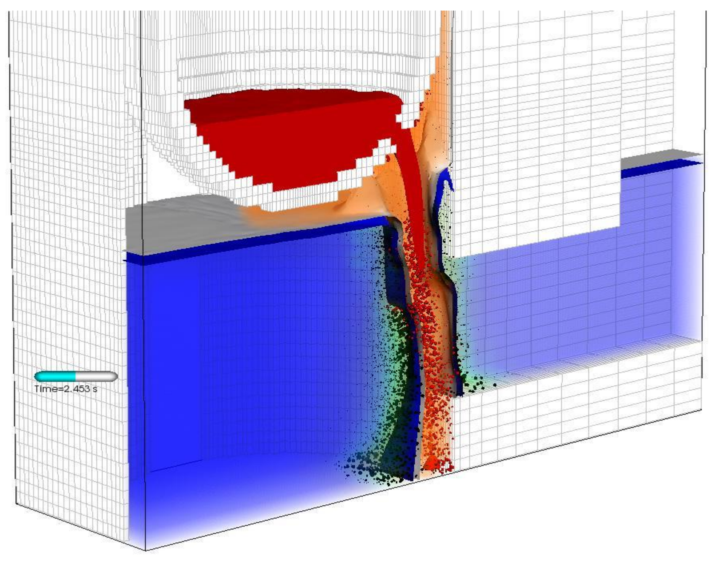
[2]CEA, DES, IRESNE, DTN, Cadarache F-13108 Saint-Paul-Lez-Durance, France

[3]CEA,DES, IRESNE, DER, Cadarache F-13108 Saint-Paul-Lez-Durance, France

Context and objectives

- **Understanding a code behavior** can be achieved by exploring the range of variation of the uncertain inputs. **Monte-Carlo methods** are useful to do so: the inputs space is **randomly sampled** and a **code run** is performed on each sampled point. However, some of these code runs may **fail to converge**. This can be due to numerical problems or suitability of the models.
- **Goal of this work**: analyse the sampled data to understand which of **the inputs** have the most influence on **code failures**.
- **Failure occurrence** can be considered as a **binary output *Z***. Hence, we consider the analysis of code failures in the general context of **sensitivity analysis**.
- Two methods are proposed to perform this particular **sensitivity analysis** [3]:
  - **Goodness-of-fit tests** that compare the initial probability distribution of each input with its conditional probability distribution knowing the code failures;
  - **Measures of the global dependence** between each input and the occurrence of code failures using the Hilbert Schmidt Independence Criterion and associated independence test.

- The **code under study** is MC3D. It computes the fuel-coolant interaction for nuclear **severe accident simulations.**



- ***d = 50 inputs*** are considered. These inputs are **uniformly distributed** around their nominal value. They are denoted $\mathrm{X} = \{X_1, ..., X_d\}$
- The binary output Z is defined as:

$$Z = \begin{cases} 1 & \text{if the code fails to converge} \\ 0 & \text{otherwise.} \end{cases}$$

- To explore the range of variation of these inputs, **2000 runs** have been performed. The samples have been obtained using a **space-filling sampling method**: **Latin Hypercube Sampling (LHS)** [4]. It improves the distribution of the design 1D-sub-projections → good regarding the **high input space dimension**. Among these code runs, ***n = 700* failed to converge.**

Method 1: Kolmogorov goodness-of fit test

- If an input has no direct effect on code failures, then the subset such that the code fails (Z = 1) is independent of the values taken by this input.

  → **Samplings such as the code fail** follow the same distributions as the **initial ones** (known).

- **We compare these two distributions**. If they are different

  → The input has a significant impact on code failure.
- **Kolmogorov-Smirnov (K-S) independence test** is used to conduct this comparison [1]. The test statistic S_n^{KS} is defined by:

$$S_n^{KS} = n^{1/2}\sup|F_n - F_0|$$

- Here, F_n is the empirical distribution function tested. F_0 is the known cumulative distribution used for comparison.

Method 2: Measure of the global independence (HSIC)

- Main idea: **measure the dependence** between each input and the code failure output Z. We use the Hilbert Schmidt Independence criterion to process this measure [2].
- **Two feature maps** $\phi\backslash\psi$ (with a Reproducing kernel Hilbert space and a kernel) are associated to X_i and Z respectively. It allows to write the generalized covariance operator as the tensor product of these centered features maps:

$$C_{X_i,Z} = \mathbb{E}[(\phi - \mathbb{E}_{X_i}(\phi)) \otimes (\psi - \mathbb{E}_Z(\psi))]$$

- Finally, the HSIC corresponds to the **square norm** of this operator:

$$HSIC(X_i, Z) = ||C_{X_i,Z}||^2$$

- Interesting property: under some conditions (characteristic kernels), **if *HSIC(Xi,Z) = 0***, then ***Xi* and *Z* are independent**. An **asymptotic test** is built around this criterion using this property [2].

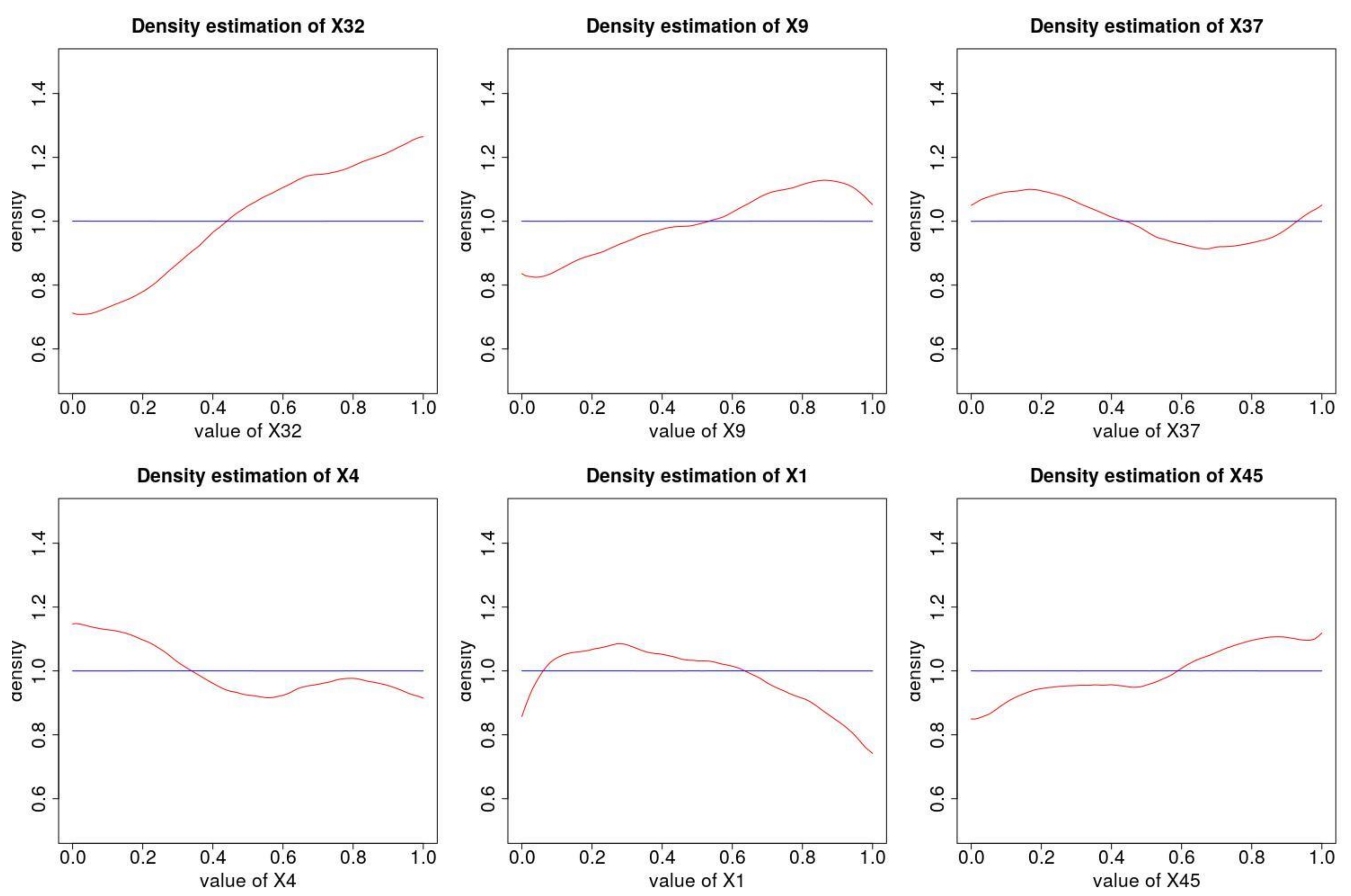
Results

We apply the **tests** associated with the **K-S and HSIC statistics** on this data set. The following table gives the **selected inputs** regarding this two tests, at level *0.1*.

input	p-value KS test	p-value HSIC test
X_{32}	0.000	0.000
X_9	0.004	0.001
X_{37}	0.006	0.021
X_4	0.007	0.038
X_1	0.016	0.012
X_{45}	0.017	0.024
X_{39}	0.020	0.001
X_{36}	0.025	0.032
X_{42}	0.036	0.047
X_{50}	0.047	0.12
X_{10}	0.051	0.052
X_{34}	0.064	0.015
X_{24}	0.075	0.234

The estimated density of the normalized marginal samples detected as failed are compared to the **initial uniform densities**.

→ Paves the way for a **physical interpretation of input influence on code failures**[3].



References:

[1] William J. Conover. *Practical Nonparametric Statistics*. New York: John Wiley & Sons, 1975.

[2] A. Gretton, O. Bousquet, A. Smola, and B. Schölkopf. *Measuring Statistical Dependence with Hilbert-Schmidt Norms.* Algorithmic Learning Theory 3734, pages 63–77, 2005.

[3] F. Hakimi, C. Brayer, A. Marrel, F. Gamboa and B. Habert. *Statistical methods for the study of computer experiments failures: Application to a fuel-coolant interaction simulation code*. Submitted to Nuclear science and engineering, 2022.

[4] M. Mckay, R. Beckman, W. Conover. *A comparison of three methods for selecting vales of input variables in the analysis of output from a computer code*. Technometrics 2, pages 239–245, 1979.