Treatment of uncertainties in multi-physics model for wind turbine asset management

Elias FEKHARI

Électricité de France R&D
Chatou, France
&
Côte d'Azur University
Nice, France

This dissertation is submitted for the degree of
Doctor of Philosophy

publicly defended on January xx, 2024 in front of the following jury:

Pr. Mireille BOSSY,	INRIA, Sophia-Antipolis	Examiner
Dr. Vincent CHABRIDON	EDF R&D, Chatou	Co-advisor
Dr. Sébastien DA VEIGA	ENSAI, Rennes	Examiner
Dr. Bertrand IOOSS	EDF R&D, Chatou	Thesis director
Dr. Anaïs LOVERA	EDF R&D, Saclay	Invite
Dr. Joseph MURÉ	EDF R&D, Chatou	Co-advisor
Pr. Franck SCHOEFS	Nantes Université, Nantes	Reviewer
Pr. Daniel STRAUB	TUM, Munich	Reviewer
Pr. Bruno SUDRET	ETH, Zürich	Examiner

Table of contents

List of figures

List of tables

Introduction

Industrial context and motivation

The shift in wind energy projects from limited onshore resources to the vast potential of offshore locations is a growing trend. Offshore wind energy offers several advantages, including more consistent winds and the ability to install larger turbines Since the installation of the first offshore wind farm in Vindeby, Denmark, in 1991, the industry has experienced rapid growth, with a total capacity of 56GW exploited worldwide in 2021. Over time, offshore wind technology has matured, resulting in significant achievements such as securing projects in Europe through "zero-subsidy bids," where electricity generated by wind farms is sold at wholesale prices.

However, despite the progress of this sector, scaling limitations emerge and numerous scientific challenges. To meet ambitious national and regional development targets, the wind energy industry must address various scaling issues, including port logistics, the demand for critical natural resources, and sustainable end-of-life processes. Furthermore, the field presents various scientific challenges that often involve coupling data with numerical simulations of physical systems and their surrounding environment. The wind energy community is focused on several objectives, including enhancing the design of floating offshore wind turbines, refining wind resource estimation techniques, and optimizing maintenance operations. Additionally, the design, installation and exploitation of these industrial assets implicate several decision-making steps, considering limited access to information. Therefore, properly modeling and treating the various uncertainties along this process proved to be a key success factor in this highly competitive industry.

Overall, the industry needs methods and techniques for uncertainty management to optimize safety margins and asset management As a wind farm project developer, the attention is first drawn to refining the wind potential of candidate sites by combining different sources of information and modeling the multivariate distribution of environmental conditions within a wind farm. In floating projects, the probabilistic design helps to define safer and more robust solutions. As a wind farm owner, another significant consideration revolves around end-of-life management. This involves evaluating three possible outcomes: extending the operating assets' lifetime, replacing current turbines with more advanced models, or dismantling and selling the wind farm. The first two solutions require assessing the current reliability of the structure and its remaining useful life. These quantitative evaluations are studied by certification bodies and insurance providers to issue exploitation permits. To deliver rigorous risk assessments, the generic *uncertainty quantification methodology* may be adopted.

Generic methodology for uncertainty quantification

Uncertainty Quantification (UQ) aims at modeling and managing uncertainties in complex systems. Over the year, generic UQ frameworks were proposed [add ref Deroc.] to quantify and analyze the relations between uncertain input factors and the systems' outcomes. UQ is particularly relevant in situations where experiments or direct observations are costly, time-consuming, or even impossible to conduct.

Computer experiments, also known as numerical experiments or simulations, play an important role in UQ. They involve the use of numerical models to simulate the behavior of a system under various conditions and parameter settings. These virtual experiments provide a cost-effective way to explore the behavior of complex systems and make robust and well-informed decisions. They enable researchers and decision-makers to gain a deeper understanding of the system dynamics, optimize designs, assess risk, and make robust predictions. As a result, uncertainty quantification has become an essential tool in wind energy, benefiting from the multiphysics numerical models simulating the behavior of wind farms interacting with their environment. Nevertheless, numerical models should be calibrated against measured data and pass validation, and verification processes [add ref] to minimize the residual modeling error. Figure 1 illustrates the UQ methodologies and the standardized usual steps encountered during a study, which are detailed hereafter:

- **Step A – Problem specification**: at this step, it is necessary to establish the system under study and construct a numerical model capable of precisely simulating its behavior. Specifying the problem also involves defining the complete set of parameters inherent to the computer model. This includes the input variables as well as determining the specific output quantity that will be generated by the numerical model;

- **Step B – Uncertainty modeling**: The objective of the second step is to identify all the sources of uncertainty impacting the input variables. Most of the time choosing a probabilistic framework, the modeling methods will depend on the available information (e.g., amount of data, input dimension);

- **Step C – Uncertainty propagation**: This step consists in propagating the uncertain inputs through the computer model, making the output uncertain. Then, the goal becomes the estimation of a quantity of interest (i.e., a statistic on the random output variable of interest). The uncertainty propagation method may differ depending on the quantity of interest targeted (e.g., central tendency, rare event);

- **Step C' – Inverse analysis**: In this additional step, a sensitivity analysis can be performed to study the role allocated to each uncertain input leading to the uncertain output;

- **Metamodeling**: Since this methodology is frequently used with computationally expensive numerical models, it becomes interesting to emulate these models using statistical models constructed from a limited number of simulations. The uncertainty quantification is then performed on the so-called "metamodel" (or surrogate model) at a reasonable computation cost.

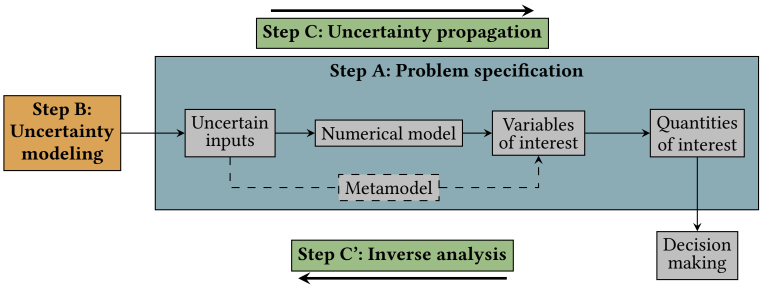

Step C: Uncertainty propagation
Step A: Problem specification
Step B: Uncertainty modeling
Uncertain inputs
Numerical model
Variables of interest
Quantities of interest
Metamodel
Decision making
Step C': Inverse analysis

Fig. 1 General uncertainty quantification framework (adapted from Ajenjo (2023))

Problem statement and outline of the thesis

[Rewrite para] A general topic of research for EDF R&D is to adapt the UQ methodology to offshore wind turbine industrial cases. However, this problem presents various specificities, raising scientific challenges. First, the numerical model studied is composed of a series of three codes, among which one is intrinsically stochastic (i.e., running twice the same numerical model with the same set of inputs results in different outputs). Second, the computational cost of these numerical models quickly requires the use of efficient techniques deployed on high-performance computers to perform UQ. Then, the probabilistic modeling tools available to model the uncertain inputs are challenged by a complex underlying dependence structure In the presence of large amounts of data describing these complex inputs, different methods to quantify and propagate the uncertainties are needed. Finally, performing a risk assessment on this case study combines all the challenges previously stated. In order to adapt the UQ framework to this industrial case, this thesis aims at answering the following questions:

Q1 *How to accurately model the complex dependence structure underlying the multivariate distribution of the environmental conditions?*

Q2 *How to perform an efficient and accurate given-data uncertainty propagation on a costly and stochastic numerical model?*

Q3 *How to couple rare event estimation with reliability-oriented sensitivity analysis?*

To intend at solving these problems, this thesis is divided into three parts. The first part gathers an introduction to UQ's state-of-the-art and a specification of the offshore wind turbine problem. The second part presents the contributions to uncertainty quantification and propagation while the third part the contributions to rare event estimation. This manuscript is divided into seven chapters, which are summarized hereafter:

Chapter 1 introduces the Treatment of uncertainties in computer experiments. Uncertainty quantification

Chapter 2 Introduction to wind turbine modeling and design

Chapter 3 Kernel-based uncertainty quantification

Chapter 4 Kernel-based central tendency estimation

Chapter 5 Kernel-based metamodel validation

Chapter 6 Nonparametric rare event estimation

Chapter 7 Sequential reliability oriented sensitivity analysis

Numerical developments

In the vain of an open-data approach, this aims at sharing the implementations developed and allows the reader to reproduce numerical results. Along this thesis, the contributions to numerical developments are summarized below:

`otkerneldesign`[1]

- This Python package generates designs of experiments based on kernel methods such as Kernel Herding and Support Points. A tensorized implementation of the algorithms was proposed, significantly increasing their performances. Additionally, optimal weights for Bayesian quadrature are provided.
- This Python package, developed in collaboration with J.Muré, is available on the platform Pypi and fully documented.

`bancs`[2]

- This Python package proposes an implementation of the "Bernstein Adaptive Nonparametric Conditional Sampling" method for rare event estimation.
- This Python package is available on the PyPI platform and is illustrated with examples and analytical benchmarks.

`ctbenchmark`[3]

- This Python package presents a standardized process to benchmark different sampling methods for central tendency estimation.
- This Python package is available on a GitHub repository with analytical benchmarks.

`copulogram`[4]

- This Python package proposes an implementation of a synthetic visualization tool for multivariate distributions.
- This Python package, developed in collaboration with V.Chabridon, is available on the Pypi platform and fully documented.

[1] Documentation: https://efekhari27.github.io/otkerneldesign/master/
[2] Repository: https://github.com/efekhari27/bancs
[3] Repository: https://github.com/efekhari27/ctbenchmark
[4] Repository: https://github.com/efekhari27/copulogram

Publications and communications

The research contributions in this manuscript are based on the following publications:

Book Chap.	E. Fekhari, B. Iooss, J. Muré, L. Pronzato and M.J. Rendas (2023). "Model predictivity assessment: incremental test-set selection and accuracy evaluation". In: *Studies in Theoretical and Applied Statistics*, pages 315–347. Springer.
Jour Pap.	E. Fekhari, V. Chabridon, J. Muré and B. Iooss (2023). "Fast given-data uncertainty propagation in offshore wind turbine simulator using Bayesian quadrature". In: *Data-Centric Engineering.*
Int. Conf	E. Fekhari, B. Iooss, V. Chabridon, J. Muré (2022). "Numerical Studies of Bayesian Quadrature Applied to Offshore Wind Turbine Load Estimation". In: *SIAM Conference on Uncertainty Quantification (SIAM UQ22)*, Atlanta, USA. (Talk)
	E. Fekhari, B. Iooss, V. Chabridon, J. Muré (2022). "Model predictivity assessment: incremental test-set selection and accuracy evaluation". In: *22nd Annual Conference of the European Network for Business and Industrial Statistics (ENBIS 2022)*, Trondheim, Norway. (Talk)
	E. Fekhari, B. Iooss, V. Chabridon, J. Muré (2022). "Efficient techniques for fast uncertainty propagation in an offshore wind turbine multi-physics simulation tool". In: *Proceedings of the 5th International Conference on Renewable Energies Offshore (RENEW 2022)*, Lisbon, Portugal. (Paper & Talk)
	E. Fekhari, V. Chabridon, J. Muré and B. Iooss (2023). "Bernstein adaptive nonparametric conditional sampling: a new method for rare event probability estimation". In: *Proceedings of the 13th International Conference on Applications of Statistics and Probability in Civil Engineering (ICASP 14)*, Dublin, Ireland. (Paper & Talk)
	E. Vanem, E. Fekhari, N. Dimitrov, M. Kelly, A. Cousin and M. Guiton (2023). "A joint probability distribution model for multivariate wind and wave conditions". In: *Proceedings of the ASME 2023 42th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2023)*, Melbourne, Australia. (Paper)
	A. Lovera, E. Fekhari, B. Jézéquel, M. Dupoiron, M. Guiton and E. Ardillon (2023). "Quantifying and clustering the wake-induced perturbations within a wind farm for load analysis". In: *Journal of Physics: Conference Series (WAKE 2023)*, Visby, Sweden (Paper)
Nat. Conf.	E. Fekhari, B. Iooss, V. Chabridon, J. Muré (2022). "Kernel-based quadrature applied to offshore wind turbine damage estimation". In: *Proceedings of the Mascot-Num 2022 Annual Conference (MASCOT NUM 2022)*, Clermont-Ferrand, France (Poster)
	E. Fekhari, B. Iooss, V. Chabridon, J. Muré (2023). "Rare event estimation using nonparametric Bernstein adaptive sampling". In: *Proceedings of the Mascot-Num 2023 Annual Conference (MASCOT-NUM 2023)*, Le Croisic, France (Talk)

Part I:

Introduction to uncertainty quantification and wind energy

Toute pensée émet un coup de dé.

S. Mallarmé

Chapter 1

Treatment of uncertainties in computer experiments

1.1 Introduction

The progress of computer simulation gradually allows the virtual resolution of more complex problems in scientific fields such as physics, astrophysics, engineering, climatology, chemistry, or biology. This domain often provides a deterministic solution to complex problems depending on several inputs. Associating a UQ analysis with these possibly nonlinear numerical models is a key element to improving the understanding of the phenomena studied. A wide panel of UQ methods has been developed over the years to pursue these studies with a reasonable computational cost.

This chapter presents the standard tools and methods from the generic UQ framework [add ref to intro], exploited later in this thesis. It is structured as follows: Section [add ref] describes the context of the model specification step; Section [add ref] presents a classification of the inputs uncertainties and the probabilistic framework to model them; Section [add ref] and xx introduce various methods to propagate the input uncertainties through the numerical model for different purposes; finally, Section [add ref] presents the main inverse methods to perform sensitivity analysis in our framework.

1.2 Black-box model specification

The uncertainty quantification studies in our framework are performed around an input-output numerical simulation model. This numerical model, or code, is hereafter considered as *black-box* since the knowledge of the underlying physics doesn't inform the UQ methods. Alternatively, one could consider *intrusive* UQ methods, introducing uncertainties within the resolution of computer simulation (see e.g., [add ref]). In practice, the numerical model might be a sequence of codes executed in series to obtain a variable of interest.

Moreover, the simulation model is in most cases deterministic, otherwise, it is qualified as intrinsically stochastic (i.e., two runs of the same model taking the same inputs return different outputs). Then, most numerical simulation presents modeling errors. In the following, it will be assumed that the numerical models passed a *validation & verification* phase, to quantify their confidence and predictive accuracy.

Formally, part of the problem specification is the definition of the set of d input variables $\mathbf{x} = (x_1, \dots, x_d)^\top$ considered uncertain (e.g., wind speed, wave period, etc.). In this thesis, the models considered will only present scalar outputs. UQ methods dedicated to other types of outputs exist (see e.g., for time series outputs Lataniotis (2019), [functional Alvaro?]). Let us then define the following numerical model:

$$\mathcal{M} : \left| \begin{array}{rcl} \mathcal{D}_{\mathbf{x}} \subseteq \mathbb{R}^d & \longrightarrow & \mathcal{D}_{\mathbf{y}} \subseteq \mathbb{R} \\ \mathbf{x} & \longmapsto & y. \end{array} \right. \tag{1.1}$$

Unlike the typical machine learning input-output dataset framework, the UQ analyst can simulate the output image of any inputs (in the input domain), using the numerical model.

However, numerical simulations often come with an important computational cost. Therefore UQ methods should be efficient and require as few simulations as possible. In this context, metamodels (or surrogate models) are statistical approximations of the costly numerical model, that can be used to perform tractable UQ. Metamodels are only built and validated on a limited number of simulations (in a *supervised learning* framework). In practice, the model specification step is often associated with the development of a *wrapper* of the code [explain wrapper], with can be deployed on a *high-perfomance computer*. Once the model is specified, a critical step of uncertainty quantification is enumerating the input uncertainties and building an associated mathematical model.

1.3 Enumerating and modeling the uncertain inputs

1.3.1 Sources of the input uncertainties

To ensure a complete risk assessment (e.g., associated with the exploitation of a wind turbine throughout its life span), the analyst should construct a list of uncertain inputs as exhaustive as possible. Even if these uncertainties might have different origins, they should all be considered jointly in the UQ study. The authors proposed to classify them for practical purposes into two groups:

- **aleatory uncertainty** regroups the uncertainties that arise from natural randomness (e.g., [add example]). From a risk management point of view, these uncertainties are qualified as *irreducible* since the industrials facing them will not be able to acquire additional information to reduce them (e.g., additional measures).

- **epistemic uncertainty** gathers the uncertainties resulting from a lack of knowledge. Contrarily to the aleatory ones, epistemic uncertainties might be reduced by investigating their origin.

Der Kiureghian and Ditlevsen (2009) offers a discussion on the relevance of this classification. They affirm that this split is practical for decision-makers to identify possible ways to reduce their uncertainties. However, this distinction should not affect the way of modeling or propagating uncertainties. [To illustrate the limits of this split, some uncertainties present both an aleatory and epistemic aspect.] In the following, the probabilistic framework is introduced to deal with uncertainties.

1.3.2 Modeling uncertain inputs with the probabilistic framework

Uncertainties are traditionally modeled with objects from the probability theory. In this thesis, the *probabilistic framework* is adopted. Alternative theories exist to mathematically model uncertainties. For example, imprecise probability theory allows more general modeling of the uncertainties. It becomes useful when dealing with very limited and possibly contradictory information (e.g., expert elicitation). The core probabilistic tools and objects are introduced hereafter.

The *probability space* (i.e., a measure space with its total measure summing to one), also called probability triple and denoted $(\Omega, \mathcal{A}, \mu)$. This mathematical concept first includes a sample space Ω, which contains a set of outcomes $\omega \in \Omega$. An *event* is defined as a set of outcomes in the sample space. Then, a σ-algebra $\mathcal{A}$ (also called event space) is a set of events. Finally, a probability function $\mu : \mathcal{A} \longrightarrow [0, 1]$, is a positive probability measure associated with an event. Most often, the choice of the probability space will not be specified. The main object will be functions defined over this probability space: random variables.

The *random vector* $\mathbf{X}$ (i.e., multivariate random variable) is a measurable function defined as:

$$\mathbf{X} : \left| \begin{array}{ccl} \Omega & \longrightarrow & \mathcal{D}_{\mathbf{x}} \subseteq \mathbb{R}^d \\ \omega & \longmapsto & \mathbf{X}(\omega) = \mathbf{x}. \end{array} \right. \tag{1.2}$$

In the following, the random vector $\mathbf{X}$ will be considered to be a squared-integrable function against the measure μ (i.e., $\int_\Omega |\mathbf{X}(\omega)|^2 \, \mathrm{d}\mu(\omega) < \infty$). Moreover, this work will focus on continuous random variables.

The *probability distribution* of the random vector $\mathbf{X}$ is the pushforward measure of μ by $\mathbf{X}$. Which is a probability measure on $(\mathcal{D}_{\mathbf{x}}, \mathcal{A})$, denoted $\mu_{\mathbf{X}}$ and defined by:

$$\mu_{\mathbf{X}}(B) = \mu(\mathbf{X} \in B) = \mu(\omega \in \Omega : \mathbf{X}(\omega) \in B), \quad \forall B \in \mathcal{A}. \tag{1.3}$$

The *cumulative distribution function* (CDF) is a common tool to manipulate random variables. It is a function $F_{\mathbf{X}} : \mathcal{D}_{\mathbf{x}} \longrightarrow [0, 1]$ defined for all $\mathbf{x} \in \mathcal{D}_{\mathbf{x}}$ as:

$$F_{\mathbf{X}}(\mathbf{x}) = \mu(\mathbf{X} \leq \mathbf{x}) = \mu(X_1 \leq x_1, \dots, X_d \leq x_d) = \mu_{\mathbf{X}}(]-\infty, x_1] \times \cdots \times]-\infty, x_d]). \tag{1.4}$$

The CDF is a positive, increasing, right-continuous function, which tends to 0 as $\mathbf{x}$ tends to $-\infty$ and to 1 as $\mathbf{x}$ tends to $+\infty$. In the continuous case, one can also define a corresponding *probability density function* (PDF) $f_{\mathbf{X}} : \mathcal{D}_{\mathbf{x}} \longrightarrow \mathbb{R}_+$ with $f_{\mathbf{X}}(\mathbf{x}) = \frac{\partial^d F_{\mathbf{X}}(\mathbf{x})}{\partial x_1 \dots \partial x_d}$.

The expected value of a random vector $\mathbb{E}[\mathbf{X}]$, also called the first moment, is a vector defined as:

$$\mathbb{E}[\mathbf{X}] = \int_\Omega \mathbf{X}(\omega) \, \mathrm{d}\mu(\omega) = \int_{\mathcal{D}_{\mathbf{x}}} \mathbf{x} f_{\mathbf{X}}(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \left(\mathbb{E}[X_1], \dots, \mathbb{E}[X_d]\right)^\top. \tag{1.5}$$

In addition, considering two random variables X_j and X_j, with $i, j \in \{1, \dots, d\}$, one can write their respective variance:

$$\mathrm{Var}(X_i) = \mathbb{E}\left[X_i - \mathbb{E}[X_i]\right], \tag{1.6}$$

and a covariance describing their joint variability:

$$\mathrm{Cov}(X_i, X_j) = \mathbb{E}\left[\left(X_i - \mathbb{E}[X_i]\right)\left(X_j - \mathbb{E}[X_j]\right)\right]. \tag{1.7}$$

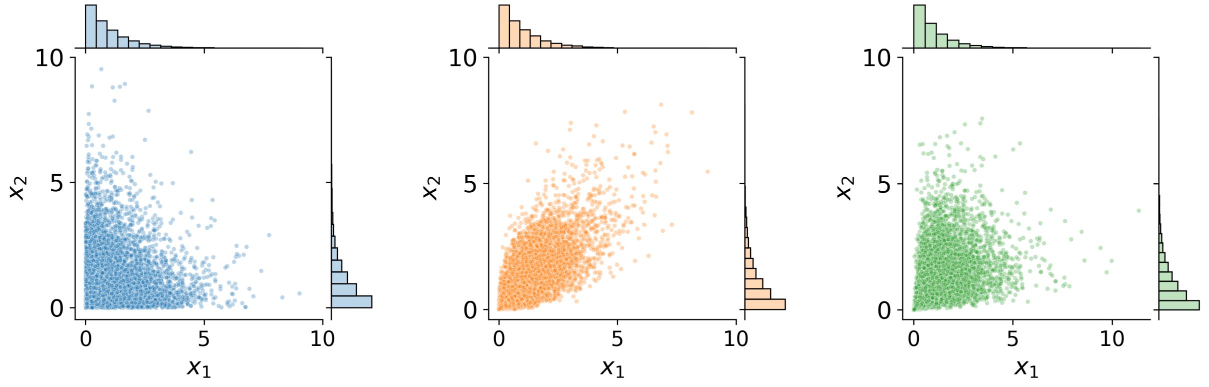
The standard deviation $\sigma_{X_j} = \sqrt{\mathrm{Var}(X_j)}$ and coefficient of variation $\delta_{X_j} = \frac{\mathrm{Var}(X_j)}{|\mathbb{E}[X_j]|}$ are two quantities directly associated to the two first moments.

[Remark on high dimension: We call high dimension anything higher than $p > 10$ and it creates issues.]

1.3.3 Joint input probability distribution

This section aims at presenting various techniques to model and infer a joint probability distribution (or multivariate distribution). It will first introduce the *copula*, a universal mathematical tool to model the dependence structure of a joint distribution. Then, a few methods to fit a joint distribution over a dataset will be mentioned. And finally, a panel of tools to evaluate the goodness of fit between a probabilistic model and a dataset will be recalled.

From a practical point of view, people tend to properly model the single effects of their input uncertainties. However, modeling the dependence structure unlying in a joint distribution is often overlooked. To illustrate the importance of this step, Fig. 1.1 represents three i.i.d samples from three bivariate distributions sharing the same single effects (e.g., here two exponential distributions) but different dependence structures. One can assume that the joint distribution is the composition of the single effects, also called marginals, and an application governing the dependence between them.



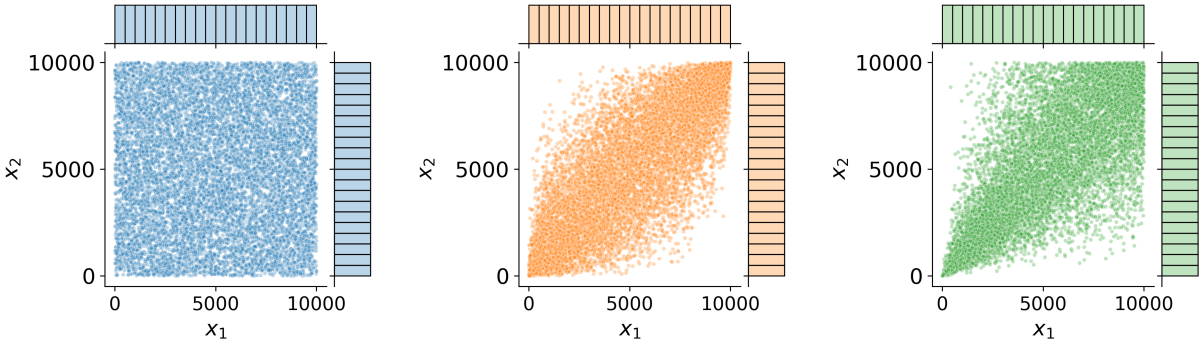
(a) Independent copula

(b) Normal copula

(c) Clayton copula

Fig. 1.1 Samples of three joint distributions with identical marginals and different dependence structures

An empirical way of isolating the three dependence structures from this example is to transform the samples in the ranked space. Let us consider a n-sized sample $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\right\} \in \mathcal{D}_{\mathbf{x}}^n$. The corresponding ranked sample is defined as: $\mathbf{R}_n = \left\{\mathbf{r}^{(1)}, \dots, \mathbf{r}^{(n)}\right\}$, where $r_j^{(l)} = \sum_{i=1}^n \mathbb{1}_{\left\{x_j^{(i)} \leq x_j^{(l)}\right\}}$, $\forall j \in \{1, \dots d\}$. Ranking a multivariate dataset allows us to isolate the dependence structure witnessed empirically. Fig. 1.2 shows the same three samples from Fig. 1.1 in the ranked space. One can first notice that the marginals are uniform since each rank is uniformly distributed. Then, the scatter plot from the distribution with independent copula (left plot) is uniform while the two others present different patterns.



(a) Independent copula

(b) Normal copula

(c) Clayton copula

Fig. 1.2 Ranked samples represented in the Fig. 1.1

A theorem states that the multivariate distribution of any random vector can be broken down into two objects (Joe, 1997). First, a set of univariate marginal distributions describing the behavior of the individual variables; Second, a function describing the dependence structure between all variables: a copula.

Theorem 1 (Sklar's theorem). *Let* $\mathbf{X} \in \mathbb{R}^d$ *be a random vector and its joint CDF* $F_{\mathbf{X}}$ *with marginals* $\{F_{X_j}\}_{j=1}^d$*, there exists a copula* $C : [0,1]^d \rightarrow [0,1]$*, such that:*

$$F_{\mathbf{X}}(x_1, \ldots, x_d) = C\left(F_{X_1}(x_1), \ldots, F_{X_d}(x_d)\right). \tag{1.8}$$

If the marginals F_{X_i} *are continuous, then this copula is unique.*

Theorem 1 expresses the joint CDF by combining marginal CDFs and a copula, which is practical for sampling joint distributions. Conversely, the copula can be defined by using the joint CDF and the marginal CDFs:

$$C(u_1, \ldots, u_d) = F_{\mathbf{X}}(F_{X_1}^{-1}(u_1), \ldots, F_{X_d}^{-1}(u_d)) \tag{1.9}$$

This equation allows us to extract a copula from a joint distribution by knowing its marginals. Additionally, copulas are invariant under creasing transformations. This property is important to understand the use of rank transformation to display the copula without the marginal effects.

[define the copula density and address the warning on the confusion.]

Identically to the univariate continuous distributions, a large catalog of families of copulas exists (e.g., independent, Normal, Clayton, Frank, Gumbel copula, etc.).

[define the independent copula, also called a product of marginals.]

To infer a joint distribution, this theorem divides the fitting problem into two independent problems: fitting the marginals and fitting the copula Provided a dataset, this framework allows the combination of a parametric (or nonparametric) fit of marginals with a parametric (or nonparametric) fit of the copula.

Appendix A details the main techniques to estimate marginal distributions. Then, Appendix B introduces different nonparametric methods to infer a copula, including the empirical Bernstein copula and the Beta copula. The adequation between a fitted probabilistic model and a dataset should be validated, therefore, appendices A and B respectively present visual and quantitative tools for goodness-if-fit evaluation.

To infer a joint distribution over a dataset, the analyst should determine a fitting strategy. Smart data visualization helps to choose the fitting methods susceptible to be relevant to the problem. The following points can be checked at this early stage:

- Is the distribution unimodal? If not, mixtures methods or nonparametric models might be required;
- Is the validity domain restrictive? If so, specific families of parametric distributions can be chosen or truncations can be applied;
- Is the dimension high? If the dimension is too high: tensorize the distribution as much as possible.
- Is the dependence structure complex? Graphically, the dataset in the ranked space gives an empirical description but some independence tests exist as well.

1.4 Central tendency uncertainty propagation

The previous section aimed at building a probabilistic model of the uncertainties considering the knowledge available. This one will introduce diverse forward propagation of uncertainty through a numerical model. This step is hereafter qualified as “global” because the analysis of the resulting output random variable will particularly focus on its central tendency (i.e., expected value and variance). This approach contrasts with the uncertainty propagation dedicated to rare event estimation, which will be introduced in the next section (e.g., for a reliability or certification problem).

The difficulties related to any uncertainty propagation mostly arise from the practical properties of the numerical model. Its potential high dimension, low regularity and nonlinearities each represent a challenge. These studies rely on a finite number of observations which depends on the computational budget the analyst can afford. This forward propagation might be a finality of the uncertainty quantification, but keep in mind that it fully stands on an accurate uncertainty modeling. Uncertainty propagation should be perceived as a standardized process with modular bricks, on which the “garbage in, garbage out” concept fully applies.

This section introduces the main methods of global uncertainty propagation. Outlining the strong links between numerical integration (i.e., Lebesgue integration or central tendency estimation) and numerical design of experiments.

1.4.1 Numerical integration

Forward uncertainty propagation aims at integrating a measurable function $g : \mathcal{D}_{\mathbf{X}} \to \mathbb{R}$ with respect to a probability measure μ. Numerical integration brings algorithmic tools to help the resolution of this probabilistic integration (i.e., Lebesgue integration).

In practice, this integral is approximated by summing a finite n-sized set of realizations $\mathbf{y}_n = \left\{g(\mathbf{x}^{(1)}), \dots, g(\mathbf{x}^{(n)})\right\}$ from a set of input samples $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\right\}$. A *quadrature* establishes a rule to select the input samples $\mathbf{X}_n$ (also called nodes), and an associated set of weights $\mathbf{w}_n = \{w_1, \dots, w_n\} \in \mathbb{R}^n$. The approximation given by a quadrature rule is defined as a weighted arithmetic mean of the realizations:

$$I_\mu(g) := \int_{\mathcal{D}_{\mathbf{X}}} g(\mathbf{x}) \mathrm{d}\mu(\mathbf{x}) \approx \sum_{i=1}^{n} w_i g\left(\mathbf{x}^{(i)}\right). \tag{1.10}$$

For a given sample size n, our goal is to find a set of tuples $\left\{\mathbf{x}^{(i)}, w_i\right\}_{i=1}^{n}$ (i.e., quadrature rule), giving the best approximation of our quantity. Ideally, the approximation quality should be fulfilled for a wide class of integrands. Most quadrature rules only depend on the measure space $(\Omega, \mathcal{A}, \mu)$, regardless of the integrand values. In the context of a costly numerical model, this property allows the analyst to massively distribute the calls to the numerical model.

This section aims at presenting the main multivariate numerical integration techniques. These methods have very different properties: some are deterministic and some are aleatory; some are sequential (or nested) some are not; some are victims of the curse of dimensionality and some are not. [A summary table of the different methods and their respective properties is proposed ADD REF TO TABLE].

Classical multivariate deterministic quadrature

Historically, quadrature methods have been developed for univariate integrals. The Gaussian rule and the Fejér-Clenshaw-Curtis rule are two univariate deterministic quadratures that will be briefly introduced.

Gaussian quadrature is a powerful univariate quadrature building together a set of irregular nodes and a set of weights. The computed weights are positive, which ensures a numerically stable rule even for large sample sizes.

Different variants of rules exist, the most famous being the Gauss-Legendre quadrature. In this case, the function g to be integrated with respect to the uniform measure on $[-1, 1]$ is approximated by Legendre polynomials. Considering the Legendre polynomial of order n, denoted l_n, the quadrature nodes ${x^{(i)}}_{i=1}^{n}$ are given by the polymonial roots. The respective weights are given by the following formula:

$$w_i = \frac{2}{\left(1 - \left(x^{(i)}\right)^2\right) \left(l'_n(x^{(i)})\right)^2}. \tag{1.11}$$

This rule guarantees a very precise approximation provided that the integrand is well-approximated by a polynomial of degree $2n-1$ or less on $[-1,1]$. This rule is deterministic but not sequential, meaning that two rules with sizes n_1 and n_2, $n_1 < n_2$ will not be nested. However, a sequential extension is proposed by the Gauss-Kronrod rule (Laurie, 1997), offering lower accuracy.

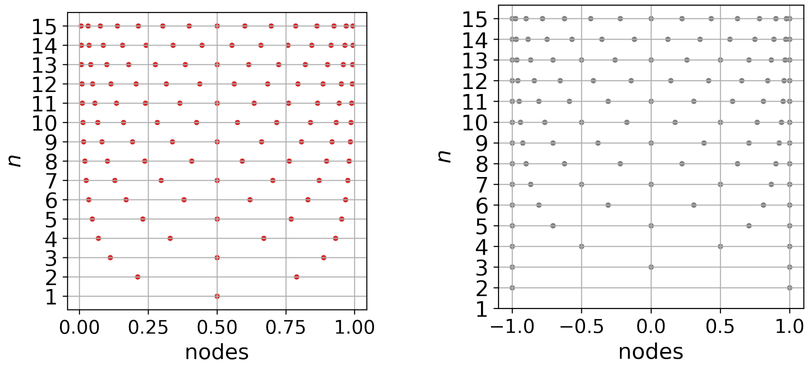
To overcome this practical drawback, Fejér then Clenshaw with Curtis proposed a nested rule with mostly equivalent accuracy as Gaussian quadrature. This method is usually presented to integrate a function with respect to the uniform measure on $[-1,1]$ and starts with a change of variables:

$$\int_{-1}^{1} g(x)\,\mathrm{d}x = \int_{0}^{\pi} g(\cos(\theta))\sin(\theta)\,\mathrm{d}\theta \tag{1.12}$$

This expression can be written as an expansion of the integrand using cosine series. Moreover, cosine series are closely related to the Chebyshev polynomials of the first kind. Fejér's "first rule" (Trefethen, 2008) relies on the Chebyshev polynomials roots as nodes $x^{(i)} = \cos(\theta^{(i+1/2)})$, and the following weights:

$$w_i = \frac{2}{n}\left(1 - 2\sum_{j=1}^{\lfloor n/2 \rfloor} \frac{1}{4j^2-1}\cos\left(j\theta^{(2i+1)}\right)\right) \tag{1.13}$$

These two univariate integration schemes are both very efficient on a wide panel of functions. Yet, Fejér-Clenshaw-Curtis is sequential and offers easy implementations, benefitting from powerful algorithms such as the *fast Fourier transform.*



(a) Gauss-Legendre

(b) Clenshaw-Curtis

Fig. 1.3 Univariate quadratures nodes ($1 \leq n \leq 15$)

Uncertainty quantification problems are rarely unidimensional, but one can build a multivariate quadrature rule by defining the tensor product (also called full grids) of univariate rules. This exhaustive approach quickly shows its practical limits as the problem's dimension increases. Alternatively, sparse multivariate quadratures (i.e., Smolyak sparse grid) explore the joint domain more efficiently.

Monte Carlo methods

Monte Carlo methods were initially developed in the 1940s to solve problems in neutronics. Ever since this frequentist techniques have been applied to the resolution of the Lebesgue integral. To integrate a function g against a measure μ, it randomly generates points following the input measure. The integral is estimated by taking the uniform arithmetic mean of the images of these nodes obtained by this random process.

This aleatory method requires to be able to generate points following a given distribution. To do so, the most common approach is to first generate a sequence of random points uniformly on $[0,1]$. These sequences mimic actual uniform randomness but are in fact generated by deterministic algorithms (also called pseudorandom number generators). Pseudorandom algorithms generate a sequence of numbers with a very large, but finite length. This sequence can be exactly repeated by fixing the same initial point, also called *pseudorandom seed*. Most programming languages use the Mersenne Twister pseudorandom generator (Matsumoto and Nishimura, 1998), offering a very long period (around 4.3×10^{6001} iterations).

Formally, the "Vanilla" Monte Carlo (sometimes called "crude" Monte Carlo) method uses a set of i.i.d samples $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\right\}$ following the joint distribution of μ. The Monte Carlo estimator of the integral is given by:

$$I_\mu(g) \approx \bar{y}_n^{MC} = \frac{1}{n} \sum_{i=1}^{n} g(\mathbf{x}^{(i)}). \tag{1.14}$$

By construction, the law of large numbers makes this estimator unbiased, however, it converges relatively slowly. Considering the images of the sample $\mathbf{X}_n$, one can also estimate the variance of the output random variable $\widehat{\sigma}_Y^2$. The variance of the Monte Carlo estimator results from a manipulation of the central tendency theorem:

$$\mathrm{Var}\left(\bar{y}_n^{MC}\right) = \frac{1}{\sqrt{n}} \mathrm{Var}\left(g(\mathbf{X})\right). \tag{1.15}$$

This estimator also comes with theoretical confidence intervals at $\alpha\%$, regardless of the output distribution:

$$I_\mu(g) \in \left[\bar{y}_n^{MC} - q_\alpha \frac{\mathrm{Var}\left(g(\mathbf{X})\right)}{\sqrt{n}}, \bar{y}_n^{MC} + q_\alpha \frac{\mathrm{Var}\left(g(\mathbf{X})\right)}{\sqrt{n}}\right], \tag{1.16}$$

where q_α is the α-quantile of the standard normal distribution. Monte Carlo presents the advantage of being a universal method, with no bias and strong convergence guarantees. Moreover, it is worth noting that its convergence properties do not depend on the dimension of the input domain. Unlike the previous multivariate deterministic quadrature, Monte Carlo doesn't suffer from the curse of dimensionality. The main limit of crude Monte Carlo is its convergence speed, making it untractable in most practical cases. More recent methods aim at keeping the interesting properties of this technique while making it more efficient. Among the *variance reduction* family of methods, let us mention importance sampling, stratified sampling (e.g., latin hypercube sampling), control variates and multi-level Monte Carlo (see Chapters 8, 9 and 10 from Owen (2013) and (Giles, 2008)).

Quasi-Monte Carlo and Koksma-Hlawka inequality

Among the methods presented so far, classical deterministic quadratures are subject to the curse of dim while Monte Carlo methods deliver contrasted performances. Quasi-Monte Carlo is a deterministic family of numerical integration schemes over $[0,1]^d$ with respect to the uniform measure on $[0,1]$. It offers powerful performances with strong guarantees by choosing nodes respecting *low discrepancy* sequences.

The discrepancy of a set of nodes (or a design) can be seen as a metric of its uniformity. The lowest the discrepancy of a design is, the “closest” it is to uniformity.

The Koksma-Hlawka theorem (Leobacher and Pillichshammer, 2014; Morokoff and Caflisch, 1995) is a fundamental result for understanding the role of the discrepancy in numerical integration.

Theorem 2 (Koksma-Hlawka). *If $g : [0,1]^d \rightarrow \mathbb{R}$ has a bounded variation (i.e., its total variation is finite), then for any design $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\right\} \in [0,1]^d$:*

$$\left| \int_{[0,1]^d} g(\mathbf{x})\mathrm{d}\mathbf{x} - \frac{1}{n}\sum_{i=1}^{n} g\left(\mathbf{x}^{(i)}\right) \right| \leq V(g) D^*(\mathbf{X}_n). \tag{1.17}$$

Where $D^*(\mathbf{X}_n)$ is the star discrepancy of the design $\mathbf{X}_n$, while $V(g)$ quantifies the complexity of the integrand, which is related to its total variation.

Where the function variation $V(g)$ in the Eq. (1.17) can formally be defined as the Hardy-Klause variation:

$$V(g) = \sum_{\mathfrak{u} \subseteq \{1,\dots,p\}} \int_{[0,1]^{\mathfrak{u}}} \left| \frac{\partial^{\mathfrak{u}} g}{\partial \mathbf{x}_{\mathfrak{u}}} (\mathbf{x}_{\mathfrak{u}}, 1) \right| \mathbf{dx}. \tag{1.18}$$

Where the L_p *star discrepancy* of a design $\mathbf{X}_n$ defined as the L_p-norm of the difference between the empirical CDF of the design $\widehat{F}_{\mathbf{X}_n}$ and the CDF of the uniform distribution $F_{\mathbf{U}}$:

$$D^*_p(\mathbf{X}_n) = \|\widehat{F}_{\mathbf{X}_n} - F_{\mathbf{U}}\|_p = \left(\int_{[0,1]^d} |\widehat{F}_{\mathbf{X}_n}(\mathbf{x}) - F_{\mathbf{U}}(\mathbf{x})|^p \, \mathbf{dx} \right)^{1/p}. \tag{1.19}$$

Additionally, the L_∞ star discrepancy can be defined from a geometric point of view. Let us consider the number of a design $\mathbf{X}_n$, falling in a subdomain $[\mathbf{0}, \mathbf{x})$ as $\#(\mathbf{X}_n \cap [\mathbf{0}, \mathbf{x}))$. Then, this empirical quantification is compared with the volume of the rectangle $[\mathbf{0}, \mathbf{x})$, noted $\mathrm{vol}([\mathbf{0}, \mathbf{x}))$. Finally, this star discrepancy is written:

$$D^*(\mathbf{X}_n) = \sup_{\mathbf{x} \in [0,1]^d} \left| \frac{\#(\mathbf{X}_n \cap [\mathbf{0}, \mathbf{x}))}{n} - \mathrm{vol}([\mathbf{0}, \mathbf{x})) \right| \tag{1.20}$$

[Figure xx empirically represents discrepancy concept] Let us point out that this star discrepancy is equivalent to the Kolmogorov-Smirnov test verifying whether the design follows a uniform distribution.
One can notice how the Koksma-Hlawka inequality dissociates the quadrature performance into a contribution from the function complexity and one from the repartition of the quadrature nodes. Knowing that the complexity of the studied integrand is fixed, this property explains the motivation to generate low-discrepancy quadratures in numerical integration.

Note that the design can also be considered as a discrete distribution (uniform sum of Dirac distributions). The discrepancy can then be expressed as a probabilistic distance between this discrete distribution and the uniform distribution. A generalized discrepancy between distributions will be presented in the [Part 2].

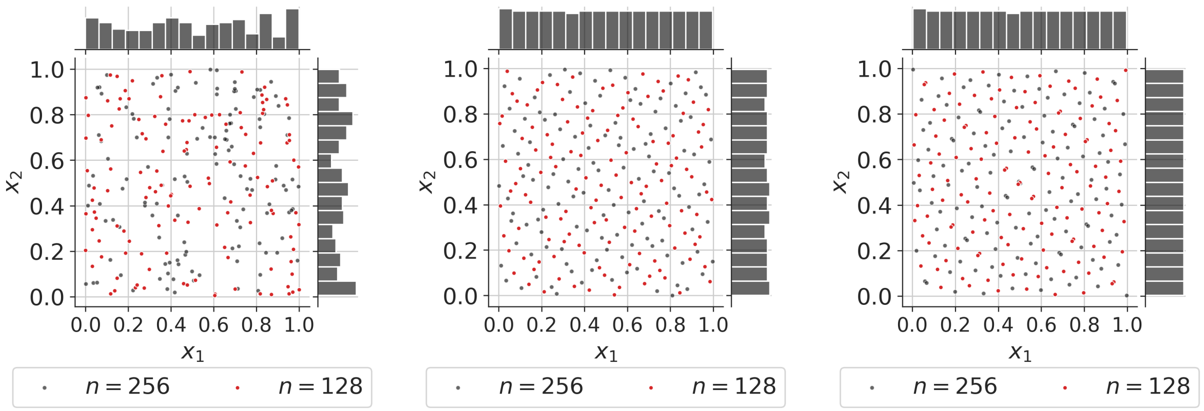
Some famous low-discrepancy sequences (e.g., van der Corput, Halton, Sobol', Faure, etc.) can offer a bounded star discrepancy $D^*(\mathbf{X}_n) \leq \frac{C\log(n)^d}{n}$, with C a constant depending on the sequence. Therefore, using these sequences as a quadrature rule with uniform weights provides the following absolute error upper bound:

$$\left| \int_{[0,1]^d} g(\mathbf{x}) \mathrm{d}\mathbf{x} - \frac{1}{n} \sum_{i=1}^{n} g\left(\mathbf{x}^{(i)}\right) \right| \leq \frac{V(g)\log(n)^d}{n} \tag{1.21}$$

The generation of these sequences doesn't necessarily require more effort than pseudo-random sampling. Chapter 15 in Owen (2013) offers an extended presentation of the ways to generate different low-discrepancy sequences. For example, the van der Corput and Halton sequences rely on congruential generators. To overcome the limits of Halton sequences, digital nets such as the famous Sobol' or Faure sequences have been developed. Sobol' sequences are in base two and have the advantage of being extensible in dimension. Note that by construction, these sequences offer significantly lower discrepancies for specific values. Typically designs with sizes equal to powers of two or power of prime numbers will be favorable. To illustrate the different patterns and properties of different methods, Fig. 1.4 represents the three designs of 256 points. Each is split into the first 128 points (in red) and the following 128 points (in black) to show to nested properties of the QMC sequences.

A quantity estimated by crude MC comes with some associated confidence. This complementary information is essential to deliver an end-to-end uncertainty quantification and misses in QMC methods. *Randomized quasi-Monte Carlo* (RQMC) is a method adding some randomness in QMC in order to compute confidence intervals while benefiting from a low variance. A specific review of the randomized (also called "scrambled") QMC is proposed by L'Ecuyer (2018). Various authors recommend the use of RQMC by default instead of QMC as a good practice. Recent works aim at exploring the use of these methods to estimate different quantities of interest (such as an expected value (Gobet et al., 2022) or a quantile (Kaplan et al., 2019))

Quasi-Monte Carlo methods easily generate powerful integration schemes. The KH inequality associates an upper bound and a convergence rate to most integrals. A randomization overlay fades the deterministic property of these designs to allow confidence interval computation. In



(a) Monte Carlo

(b) Halton sequence

(c) Sobol' sequence

Fig. 1.4 Nested Monte Carlo and quasi-Monte Carlo designs ($n = 256$)

the following, sampling techniques are presented from the numerical *design of experiments* point of view. Even if the finality might look different from the previous numerical integration, it shares many methods and concepts.

1.4.2 Numerical design of experiments

The numerical design of experiments aims at exploring uniformly the input domain, e.g., to build the learning set of a regression model, or to initialize a multi-start optimization strategy. A design of experience (also simply called design) is then qualified as *space-filling* when it properly covers a domain. As well as in integration, a design of experiments allows propagating uncertainties through a numerical model (or an actual experiment from a laboratory test bench). However, a difference comes from the fact that this community often works with designs of very limited sizes. Users of designs of experiments might also need to build designs with various properties.

- Some might be interested in the sequentiality of a sampling method, to eventually add new points as they get a computational budget extension.
- Some might request a sampling method conserving its properties in any sub-domains. This second property can be useful to reduce the problem's dimension by dropping a few unimportant marginals.

Different metrics are commonly used to quantify how space-filling a design of experiments is. The previously introduced different types of discrepancies are space-filling metrics. Other types of space-filling metrics rely on purely geometrical considerations.

This section will first define some space-filling metrics. Secondly, the *latin hypercube sampling* (LHS) will be introduced as a variance-reduction that became popular in this community. Finally, a general discussion on uncertainty propagation with respect to non-uniform measures will be presented.

Space-filling metrics and properties

Space-filling criteria are key to evaluating designs and are often used in their construction to optimize their performances. In the previous section, the star discrepancy was introduced as a distance of a finite design to uniformity. However, the L_∞ star discrepancy is hard to estimate, fortunately, Warnock (1972) elaborated an explicit expression specific to the L_2 star discrepancy:

$$\left[D_2^*(\mathbf{X}_n)\right]^2 = \frac{1}{9} - \frac{2}{n}\sum_{i=1}^{n}\prod_{l=1}^{d}\frac{(1-x_l^{(i)})}{2} + \frac{1}{n^2}\sum_{i,j=1}^{n}\prod_{l=1}^{d}\left[1-\max(x_l^{(i)}, x_l^{(j)})\right]. \tag{1.22}$$

One can notice that this expression is similar to the Cramér-von Mises test statistic. Even if this expression is tractable, Fang et al. (2018) detailed its limits. First, the star L_2 discrepancy generates designs that are not robust to projections in sub-spaces. Then, this metric is not invariant in rotation and reflection. Finally, by construction, L_p discrepancies give a special role to the point $\mathbf{0}$ by anchoring the box $[\mathbf{0}, \mathbf{x})$.

Two improved criteria were proposed by Hickernell (1998) with the *centered* L_2 *discrepancy* and the *wrap-around* L_2 *discrepancy*. Those are widely used in practice since they solve the previous limits while satisfying the Koksma-Hlawka inequality with a modification of the total variation. Let us introduce the formula of the centered L_2 discrepancy:

$$\begin{aligned} CD_2^*(\mathbf{X}_n) = \left(\frac{13}{12}\right)^d &- \frac{2}{n}\sum_{i=1}^{n}\prod_{l=1}^{d}\left(1+\frac{1}{2}|x_l^{(i)}-0.5| - \frac{1}{2}|x_l^{(i)}-0.5|^2\right) \\ &+ \frac{1}{n^2}\sum_{i,j=1}^{n}\prod_{l=1}^{d}\left(1+\frac{1}{2}|x_l^{(i)}-0.5| + \frac{1}{2}|x_l^{(j)}-0.5| - \frac{1}{2}|x_l^{(i)}-x_l^{(j)}|\right). \end{aligned} \tag{1.23}$$

As an alternative to discrepancies, many geometrical criteria exist to assess a space-filling design. The most common way to do so is to maximize the minimal distance among the pairs of Euclidian distances between the points of a design. The criterion to maximize is then simply called the *minimal distance* of a design (denoted ϕ_{min}). For numerical reasons, the ϕ_p criterion is often used instead of the minimal distance. The following ϕ_p criterion converges towards the minimum distance as $p \geq 1$ tends to infinity:

$$\phi_{\min}(\mathbf{X}_n) = \min_{i \neq j} ||\mathbf{x}^{(i)} - \mathbf{x}^{(j)}||_2, \qquad \phi_p(\mathbf{X}_n) = \sum_{i=1}^{j}\sum_{j=1}^{n}\left(|x^{(i)} - x^{(j)}|^{-p}\right)^{\frac{1}{p}}. \tag{1.24}$$

More space-filling criteria are reviewed in Abtini (2018). These space-filling metrics are widely used to optimize a different sampling technique.

Latin hypercube sampling

The LHS is a method introduced in 1979 (Mckay et al., 1979), initially for numerical integration. This stratified sampling technique forces the distribution of each sub-projection of a bounded domain to be as uniform as possible To do so, for a n-sized design, each marginal's domain is divided into n identical segments. This creates a regular grid of n^d squared cells over the domain.

An LHS design does not allow more than one point within a segment. That way, a new LHS can be built as a permutation of the marginals of an existing LHS. Inside each selected cell from the grid, the point can be placed in the center or randomly.

Various contributions provided first a variance, then a central limit theorem to the LHS (Koehler and Owen, 1996). Identically to the Monte Carlo variance[add pointer], LHS variance can be expressed as:

$$\mathrm{Var}\left(\bar{y}_n^{LHS}\right) = \frac{1}{\sqrt{n}} \mathrm{Var}\left(g(\mathbf{X})\right) - \frac{C}{n} + o\left(\frac{1}{n}\right). \tag{1.25}$$

Where C is a positive constant, showing that the LHS usually reduces the variance for numerical integration. Because of its stratified structure, LHS can generate poor designs from a space-filling point of view (see e.g., Figure [add diagonal design]). The following section presents various methods aiming at optimizing these designs.

Optimized Latin hypercube sampling

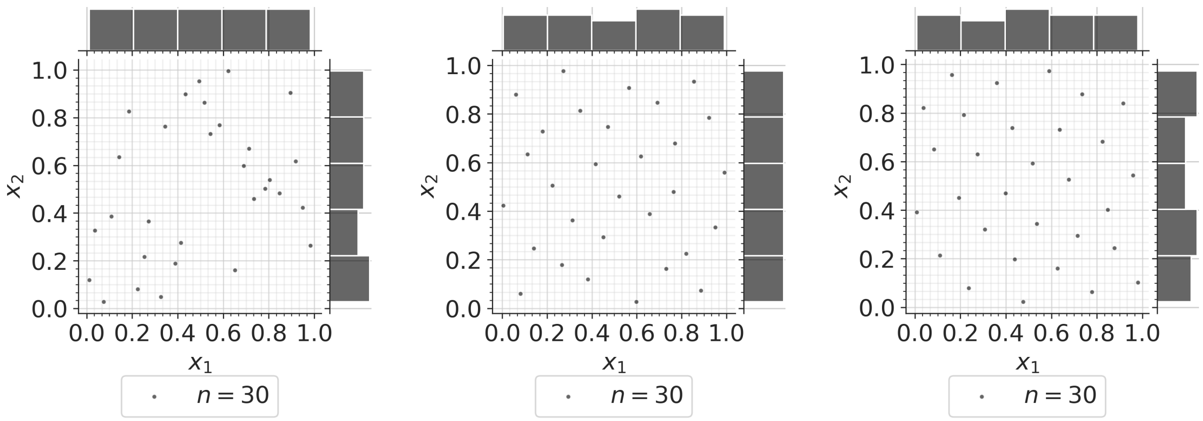
To improve the space-filling property of LHD, it is common to add an optimization step. The goal of this optimization is to improve a space-filling criterion by generating LHD from permutations of an initial LHD. Damblin et al. (2013) reviews LHS optimization using different discrepancy criteria and subprojection properties. This optimization can be performed by different algorithms, such as the stochastic evolutionary algorithm or simulated annealing. The results from this work show that LHD optimized by L_2 centered or wrap-around discrepancies offer strong robustness to two-dimensional projections. It also shows that these designs keep this property for dimensions larger than 10, while scrambled Sobol' sequences lose it.

More recent work developed different ways to get optimized LHD. Let us mention the maximum projection designs from Joseph et al. (2015) and the uniform projection designs from Sun et al. (2019). [Add a sentence to explain the concept]

1.4.3 Summary and discussion

A wide panel of sampling techniques exists for numerical integration or design of experiments purposes. In both cases, the studied domain was bounded and the targeted measure was uniform. However, uncertainty propagation is often performed on complex input distributions, with possibly unbounded domains. In uncertainty quantification, this step might be referred to as the estimation of the output random variable's central tendency (i.e., its mean and variance). Central tendency estimation is a numerical integration with respect to any input distribution, also named *probabilistic integration* by Briol et al. (2019).

To generate i.i.d samples following any distribution, one may use *inverse transform* sampling. This method first generates a sample in the unit hypercube, then, the inverse CDF function (i.e.,



(a) Poorly space-filling LHS (b) L_2 centered optimized LHS (c) ϕ_p optimized LHS

Fig. 1.5 Latin hypercube designs with poor and optimized space-filling properties ($n = 8$)

quantile function) is applied on marginals. Finally, possible dependence effects can be added using the Sklar theorem Eq. (1).

One may wonder if the properties from the uniform design are conserved after this nonlinear transformation. Li et al. (2020) explores this question from a discrepancy point of view. [add conclusions]

Let us also remark that, depending on the distribution, inverse mapping is not always possible. [give an example] As an alternative, the *acceptance-rejection* method is more versatile. [pros and cons? QMC is not suited for the acceptance-rejection method]

[Markov Chain Monte Carlo: when it is hard to sample from the distribution.]

[Add the summary table with the properties of each methods]

1.5 Reliability-oriented uncertainty propagation

1.5.1 Problem formalization

Limit-state function, failure event and domain

Risk measures [Failure probability, quantile, super-quantile]

1.5.2 Rare event estimation methods

[Why are the previous sampling methods not suited for rare events?]

FORM/SORM

Monte Carlo

Importance sampling

Adaptive sampling (SS/NAIS/IS-CE/Moving particles)

1.6 Sensitivity analysis

1.6.1 Global sensitivity analysis

1.6.2 Reliability-oriented sensitivity analysis

1.7 Metamodeling

Note that the calibration error is often larger than the metamodeling error.

1.7.1 Global metamodel

1.7.2 Contour finding for rare-event estimation

1.8 Conclusion

Chapter 2

Introduction to wind turbine modeling and design

2.1 Introduction

2.2 Wind turbine modeling

2.2.1 Synthetic wind generation [TurbSim, Kaimal spectrum]

2.2.2 Synthetic wave generation

2.2.3 Aerodynamic interactions

2.2.4 Servo-Hydro-Aero-Elastic wind turbine simulation [DIEGO]

2.2.5 Soil modeling

2.2.6 Wake modeling [FarmShadow]

2.3 Recommended design practices

2.3.1 Design load cases

2.3.2 Dynamic response design

2.3.3 Fatigue response design

2.4 Uncertain inputs

2.4.1 Environmental inputs

2.4.2 System inputs

2.4.3 Probabilistic fatigue assessment

2.5 Conclusion

References

Abtini, M. (2018). *Plans prédictifs à taille fixe et séquentiels pour le krigeage.* PhD thesis, Ecole Centrale Lyon.

Ajenjo, A. (2023). *Info-gap robustness assessment of reliability evaluations for the safety of critical industrialsystems.* PhD thesis, Université Bourgogne Franche-Comté.

Briol, F., Oates, C., Girolami, M., Osborne, M., and Sejdinovic, D. (2019). Probabilistic Integration: A Role in Statistical Computation? *Statistical Science*, 34:1 – 22.

Chabridon, V., Balesdent, M., Perrin, G., Morio, J., Bourinet, J.-M., and Gayton, N. (2021). Global reliability-oriented sensitivity analysis under distribution parameter uncertainty. *Mechanical Engineering under Uncertainties: From Classical Approaches to Some Recent Developments*, pages 237–277.

Damblin, G., Couplet, M., and Iooss, B. (2013). Numerical studies of space-filling designs: Optimization of Latin Hypercube Samples and subprojection properties. *Journal of Simulation*, 7.

Der Kiureghian, A. and Ditlevsen, O. (2009). Aleatory or epistemic? Does it matter? *Structural Safety*, 31(2):105–112.

Fang, K., Liu, M.-Q., Qin, H., and Zhou, Y.-D. (2018). *Theory and application of uniform experimental designs*, volume 221. Springer.

Giles, M. (2008). Multilevel Monte Carlo Path Simulation. *Operations Research*, 56:607–617.

Gobet, E., Lerasle, M., and Métivier, D. (2022). Mean estimation for randomized quasi Monte Carlo method.

Hickernell, F. (1998). A generalized discrepancy and quadrature error bound. *Mathematics of computation*, 67(221):299–322.

Joe, H. (1997). *Multivariate Models and Multivariate Dependence Concepts.* Chapman and Hall.

Joe, H. and Kurowicka, D. (2011). *Dependence modeling: vine copula handbook.* World Scientific.

Joseph, V., Gul, E., and Ba, S. (2015). Maximum projection designs for computer experiments. *Biometrika*, 102.

Kaplan, Z., Li, Y., Nakayama, M., and Tuffin, B. (2019). Randomized quasi-Monte Carlo for quantile estimation. In *2019 Winter Simulation Conference (WSC)*, pages 428–439.

Koehler, J. and Owen, A. (1996). 9 Computer experiments. In *Design and Analysis of Experiments*, volume 13 of *Handbook of Statistics*, pages 261–308. Elsevier.

Lasserre, M. (2022). *Apprentissages dans les réseaux bayésiens à base de copules non-paramétriques.* PhD thesis, Sorbonne Université.

Lataniotis, C. (2019). *Data-driven uncertainty quantification for high-dimensional engineering problems.* PhD thesis, ETH Zürich.

Laurie, D. (1997). Calculation of Gauss-Kronrod quadrature rules. *Mathematics of Computation*, 66(219):1133–1145.

L'Ecuyer, P. (2018). *Randomized quasi-Monte Carlo: An introduction for practitioners.* Springer.

Leobacher, G. and Pillichshammer, F. (2014). *Introduction to quasi-Monte Carlo integration and applications.* Springer.

Li, Y., Kang, L., and Hickernell, F. (2020). Is a transformed low discrepancy design also low discrepancy? *Contemporary Experimental Design, Multivariate Analysis and Data Mining: Festschrift in Honour of Professor Kai-Tai Fang*, pages 69–92.

Matsumoto, M. and Nishimura, T. (1998). Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1):3–30.

Mckay, M., Beckman, R., and Conover, W. (1979). A Comparison of Three Methods for Selecting Vales of Input Variables in the Analysis of Output From a Computer Code. *Technometrics*, 21:239 – 245.

Morokoff, W. J. and Caflisch, R. E. (1995). Quasi-Monte Carlo Integration. *Journal of Computational Physics*, 122(2):218–230.

Owen, A. (2013). *Monte Carlo theory, methods and examples.*

Sancetta, A. and Satchell, S. (2004). The Bernstein copula and its applications to modeling and approximations of multivariate distributions. *Econometric Theory*, 20(3):535–562.

Segers, J., Sibuya, M., and Tsukahara, H. (2017). The empirical beta copula. *Journal of Multivariate Analysis*, 155:35–51.

Sun, F., Wang, Y., and Xu, H. (2019). Uniform projection designs. *The Annals of Statistics*, 47:641 – 661.

Trefethen, L. (2008). Is Gauss quadrature better than Clenshaw–Curtis? *SIAM review*, 50(1):67–87.

Warnock, T. (1972). Computational investigations of low-discrepancy point sets. In *Applications of number theory to numerical analysis*, pages 319–343. Elsevier.

Appendix A

Univariate distribution fitting

This appendix recalls the main methods to infer a univariate distribution considering a n-sized i.i.d sample $X_n = \left\{x^{(1)}, \dots, x^{(n)}\right\} \in \mathbb{R}^n$. The goal is to use this finite set of observations of the random variable X to approach its underlying distribution by an estimated distribution. The inference techniques are split into two main groups, the methods assuming that the underlying distribution belongs to a family of parametric distributions are called parametric. Otherwise, the fitting method falls into the nonparametric group. Nonparametric methods often require a larger amount of data but allow more flexibility. In fact, nontrivial distributions (e.g., multimodal) might be easier to model using nonparametric approaches. To assess the quality of this estimation regarding the sample, a panel of goodness-of-fit methods are proposed [add ref], this appendix recalls a few of them. Note that the following tools can be used to estimate the marginals of a multivariate distribution.

A.1 Main parametric methods

Moments method

The moments method aims at looking for a parametric distribution with density $f_X(\boldsymbol{\theta})$, whose first moments (e.g., $m(\boldsymbol{\theta})$ and $\sigma^2(\boldsymbol{\theta})$) match the empirical moments of the sample X_n (e.g., $\widehat{m}_{X_n}$ and $\widehat{\sigma}^2$). After computing the empirical moments:

$$\widehat{m}_n = \frac{1}{n} \sum_{i=1}^{n} x^{(i)}, \qquad \widehat{\sigma}_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} \left(x^{(i)} - \widehat{m}_{X_n}\right)^2, \tag{A.1}$$

one can solve the system of equations $\left(m(\boldsymbol{\theta}) = \widehat{m}_n;\ \sigma^2(\boldsymbol{\theta}) = \widehat{\sigma}_n^2\right)$ to determine the optimal set of parameters $\boldsymbol{\theta}$ in this situation. Some families of distributions are more suited to this method (i.e.,) because of the analytical expression of their moments. Moreover, this technique is sensitive to the possible biases in the estimation of the sample moments.

Maximum likelihood estimation

Maximum likelihood estimation (MLE) is a popular alternative the moments method. Similarly, it aims at maximizing a given correspondence metric between the dataset X_n and a parametric distribution with density $f_X(\boldsymbol{\theta})$. This metric is the *likelihood* function, defined as:

$$\mathcal{L}(\boldsymbol{\theta}|X_n) = \prod_{i=1}^{n} f_X(x^{(i)}; \boldsymbol{\theta}), \tag{A.2}$$

with the PDF taking the set of parameters $\boldsymbol{\theta}$ written: $f_X(x^{(i)}; \boldsymbol{\theta})$. For numerical reasons, the optimization is often performed on the natural logarithm of the likelihood function, called *log-likelihood*. The goal is then finding the optimal vector $\widehat{\boldsymbol{\theta}}^*$ of parameters minimizing the following expression:

$$\widehat{\boldsymbol{\theta}}^* = \underset{\boldsymbol{\theta} \in \mathcal{D}_{\boldsymbol{\theta}}}{\arg\min} \left(- \sum_{i=1}^{n} \ln(f_X(x^{(i)}; \boldsymbol{\theta})) \right). \tag{A.3}$$

Remark that the quick analytical results from the moment method can be used as a starting point of the MLE optimization. [Asymptotic behaviors of this method are decribed in: add ref] [This method can be applied to censored data in the field of survival analysis. Add ref]

Example 1. Considering a small set of observations $X_n = \{1, 2, 3, 4, 6\}$, the following figure xx represents

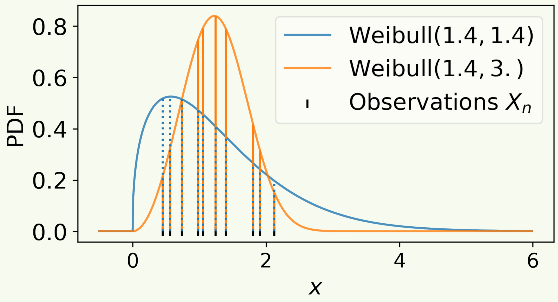


Fig. A.1 Adequation of two different Weibull models using their likelihood with a sample of observations (black crosses).

A.2 Main nonparametric methods

Empirical CDF and histogram

The empirical CDF is a cumulative stair-shaped representation of the sorted sample X_n:

$$\widehat{F}_X(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\{x \geq x^{(i)}\}}. \tag{A.4}$$

A histogram consists in sorting and gathering the observations in a sample X_n into a finite number of categories. These categories are called bins and each regroups the same number of observations (identical binwidth). The number of bins is the only tunning parameter of this method. Its definition has a great impact on the visual consistency of the plot, therefore, many rules exist to define it. Note that the empirical CDF can be seen as a cumulative histogram with the number of bins equal to the number of observations.

Kernel density estimation

Kernel density estimation (KDE) is a nonparametric method, it estimates a PDF by weighing a sample of observations X_n with kernels. After setting a kernel $k : \mathbb{R} \longrightarrow \mathbb{R}_+$ and a scaling parameter $h > 0$, also called bandwidth, the kernel density estimator is defined as:

$$\widehat{f}_X(x) = \frac{1}{nh} \sum_{i=1}^{n} k\left(\frac{x - x^{(i)}}{h}\right) \tag{A.5}$$

Different types of kernels are used for KDE, such as the uniform, triangular, squared exponential or Epachnikov. The choice of bandwidth results in a bias-variance trade-off, that has been extensively discussed in the literature [add ref].

Example 2. Considering a small set of observations $X_n = \{1, 2, 3, 4, 6\}$, the following figure xx represents three fits obtained by.

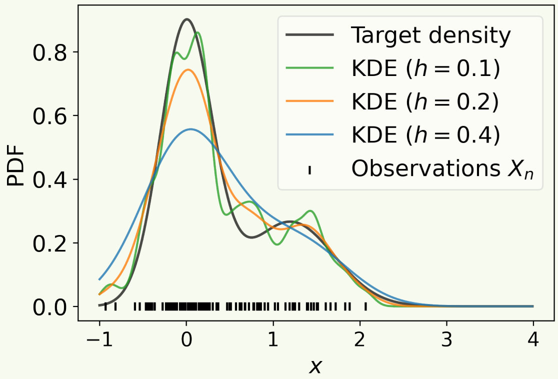


Fig. A.2 Fit of a bimodal density by KDE using different tuning parameters.

Main goodness-of-fit methods

Penalized likelihood criteria

Two quantitative goodness-of-fit criteria are commonly used to assess parametric inference: the *Akaike information criterion* (AIC) and the *Bayesian information criterion* (BIC). The likelihood as a goodness-of-fit criterion should only be applied to the same family of distributions. Otherwise, the comparison would unfairly advantage distributions with a large number of degrees of freedom. The two following criteria are metrics based on the likelihood with a correction related to the number of degrees of freedom of the distribution. Moreover, let us remind that more flexible models will require more data to provide a robust estimation.

The AIC and BIC are expressed as follows:

$$\mathrm{AIC} = \frac{-2\ln\left(\mathcal{L}(\boldsymbol{\theta}|X_n)\right)}{n} + \frac{2q}{n}, \qquad \mathrm{BIC} = \frac{-2\ln\left(\mathcal{L}(\boldsymbol{\theta}|X_n)\right)}{n} + \frac{q\ln(n)}{n}, \tag{A.6}$$

with the likelihood $\mathcal{L}(\boldsymbol{\theta}|X_n)$ and the number of distribution's number degrees of freedom denoted q. The second term adds a penalty depending on the number of parameters. The best inference will be given by the model with the smallest AIC or BIC. Note that an additional correction can be applied in a small data context.

Kolmogorov-Smirnov adequacy test

Quantile-quantile plot

The quantile-quantile plot (also called QQ-plot) is a graphical tool providing a qualitative check of the goodness of fit. It compares the CDF of the fitted model with the empirical CDF of the sample X_n. To do so, it represents a scatterplot of the empirical quantiles (i.e., the ranked observations), against the quantiles of the fitted model at the levels $\left\{\alpha^{(i)}\right\}_{i=1}^{n} = \left\{\widehat{F}_X(x^{(i)})\right\}_{i=1}^{n}$. The following [figure xx] is a QQ-plot of the model fitted in [Example xx]. The closer the scatter plot gets to the first bisector line the best the fit is.

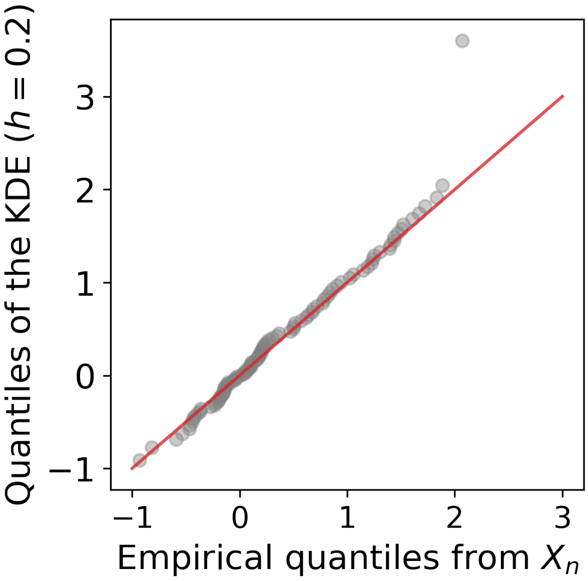


Fig. A.3 QQ-plot between the data from Example 2 and a KDE model.

Appendix B

Nonparametric copula estimation

[update CDF notations]

[Change EBC notations using h for the polynomial orders]

When the distribution's dimension is higher than two, one can perform a parametric fit using vine copulas (Joe and Kurowicka, 2011), implying the choice of multiple types of parametric copulas. Otherwise, nonparametric fit by multivariate kernel density estimation (KDE) presents a computational burden as soon as the dimension increases (Chabridon et al., 2021). Since univariate marginals are usually well-fitted with nonparametric tools (e.g., KDE), let us introduce an effective nonparametric method for copula fitting.

B.1 Empirical copula

In practice, considering a sample $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\right\} \in \mathbb{R}^{np}$ and the associated ranked sample $\mathbf{R}_n = \left\{\mathbf{r}^{(1)}, \dots, \mathbf{r}^{(n)}\right\}$, the corresponding empirical copula writes:

$$C_n(\mathbf{u}) := \frac{1}{n} \sum_{i=0}^{n} \prod_{j=1}^{p} \mathbb{1}\left\{\frac{r_j^{(i)}}{n} \leq u_j\right\}, \tag{B.1}$$

with $\mathbf{u} = (u_1, \dots, u_d) \in [0,1]^d$. In the following, the polynomial order is set as equal in each dimension: $\{m_i = m\}_{j=1}^{d}$.

B.2 Empirical Bernstein & Beta copula

Copulas are continuous and bounded functions defined on a compact set (the unit hypercube). Bernstein polynomials allow us to uniformly approximate as closely as desired any continuous and real-valued function defined on a compact set (Weierstrass approximation theorem). Therefore, they are good candidates to approximate unknown copulas. This concept was introduced as *empirical Bernstein copula* (EBC) by Sancetta and Satchell (2004) for applications in economics and risk management. Later on, Segers et al. (2017) offered further asymptotic studies. Formally, the multivariate Bernstein polynomial for a function $C : [0,1]^d \longrightarrow \mathbb{R}$ on a grid over the unit hypercube $G := \left\{\frac{0}{m_1}, \dots, \frac{m_1}{m_1}\right\} \times \dots \times \left\{\frac{0}{m_d}, \dots, \frac{m_d}{m_d}\right\}, \mathbf{m} = (m_1, \dots, m_d) \in \mathbb{N}^d$, writes:

$$B_{\mathbf{m}}(C)(\mathbf{u}) := \sum_{t_1=0}^{m_1} \cdots \sum_{t_d=0}^{m_d} C\left(\frac{t_1}{m_1}, \dots, \frac{t_d}{m_d}\right) \prod_{j=1}^{d} P_{m_j, t_j}(u_j), \tag{B.2}$$

with $\mathbf{u} = (u_1, \dots, u_d) \in [0,1]^d$, and the Bernstein polynomial $P_{m,t}(u) := \frac{t!}{m!(t-m)!} u^m (1-u)^{t-m}$. Notice how the grid definition implies the polynomial's order. When C is a copula, then $B_{\mathbf{m}}(C)$ is called "Bernstein copula". Therefore, the empirical Bernstein copula is an application of the Bernstein polynomial in Eq. (B.2) to the so-called "empirical copula". [add a sentence to mean to refer to the previous subsection]

Theoretically, the tuning parameter can be optimized to minimize an "Mean Integrated Squared Error" (MISE), leading to a bias-variance tradeoff. Formally, the MISE of the empirical Bernstein copula $B_{\mathbf{m}}(C_n)$ is defined as follows:

$$\mathbb{E}\left[\|B_{\mathbf{m}}(C_n) - C\|_2^2\right] = \mathbb{E}\left[\int_{\mathbb{R}^d} \left(B_{\mathbf{m}}(C_n)(\mathbf{u}) - C(\mathbf{u}) d\mathbf{u}\right)^2\right]. \tag{B.3}$$

Then, Sancetta and Satchell (2004) prove in their Theorem 3 that:

- $B_{\mathbf{m}}(C_n)(\mathbf{u}) \longrightarrow C(\mathbf{u})$ for any $u_j \in \,]0,1[$ if $\frac{m^{d/2}}{n} \longrightarrow 0$, when $m, n \longrightarrow \infty$.
- The optimal order of the polynomial in terms of MISE is: $m \lesssim m_{\text{IMSE}} = n^{2/(d+4)}, \forall u_j \in \,]0,1[$. The sign $\lesssim$ means "less than or approximately".

Let us remark that in the special case $m = n$, also called the "Beta copula" in Segers et al. (2017), the bias is very small while the variance gets large. To illustrate the previous theorem, Lasserre (2022) represents the evolution of the m_{IMSE} for different dimensions and sample sizes (see Fig. B.1). In high dimension, the values of m_{IMSE} tend towards one, which is equivalent to the independent copula. Therefore, high-dimensional problems should be divided into a product of smaller problems on which the EBC is tractable. Provided a large enough learning set $\mathbf{X}_n$, KDE fitting of marginals combined with EBC fitting of the copula delivers good results even on complex dependence structures. Moreover, EBC provides an explicit expression, making a Monte Carlo generation of i.i.d. samples simple.

B.3 Goodness-of-fit

[Mention the vine copulas and how we want to only use nonparametric methods here.]

[Tails correlation / Kendall plot]

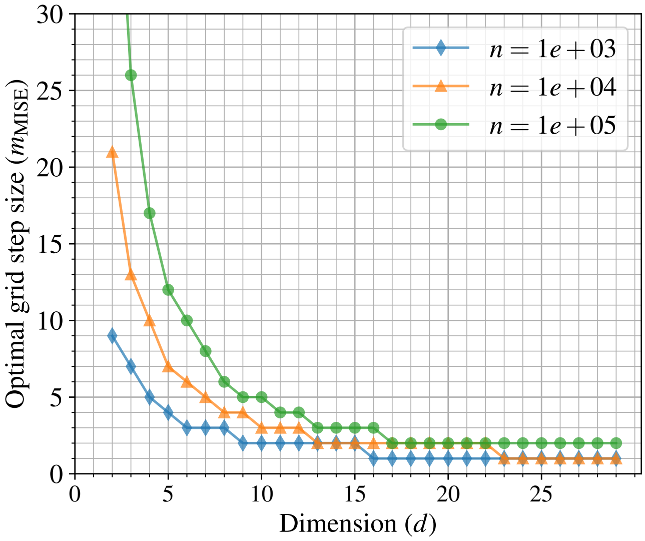


Fig. B.1 Evolution of m_{IMSE} for different dimensions and sample sizes.

Appendix C

Rare event estimation algorithms

Appendix D

Résumé étendu de la thèse