Entropy-based closure for probabilistic learning on manifolds

Christian Soize, Roger Ghanem, C. Safta, X. Huan, Z.P. P Vane, J. Oefelein, G. Lacaze, H.N. N Najm, Q. Tang, X. Chen

Entropy-based closure for probabilistic learning on manifolds

C. Soize[a,*], R. Ghanem[b], C. Safta[c], X. Huan[c], Z. P. Vane[c], J. Oefelein[c], G. Lacaze[c], H. N. Najm[c], Q. Tang[d], X. Chen[d]

[a] *Université Paris-Est Marne-la-Vallée, MSME UMR 8208 CNRS, 5 bd Descartes, 77454 Marne-La-Vallée, France*
[b] *University of Southern California, 210 KAP Hall, Los Angeles, CA 90089, United States*
[c] *Sandia National Laboratories, 7011 East Avenue, Livermore, CA 99551, United States*
[d] *Lawrence Livermore National Laboratory, Livermore, CA, United States*

Abstract

In a recent paper, the authors proposed a general methodology for probabilistic learning on manifolds. The method was used to generate numerical samples that are statistically consistent with an existing dataset construed as a realization from a non-Gaussian random vector. The manifold structure is learned using diffusion manifolds and the statistical sample generation is accomplished using a projected Itô stochastic differential equation. This probabilistic learning approach has been extended to polynomial chaos representation of databases on manifolds and to probabilistic nonconvex constrained optimization with a fixed budget of function evaluations. The methodology introduces an isotropic-diffusion kernel with hyperparameter ε. Currently, ε is more or less arbitrarily chosen. In this paper, we propose a selection criterion for identifying an optimal value of ε, based on a maximum entropy argument. The result is a comprehensive, closed, probabilistic model for characterizing data sets with hidden constraints. This entropy argument ensures that out of all possible models, this is the one that is the most uncertain beyond any specified constraints, which is selected. Applications are presented for several databases.

Keywords: Statistical learning, Probabilistic learning, Concentration of probability, Measure concentration, Probability distribution on manifolds, Random sampling generator, MCMC generator, Diffusion maps, Smoothing parameter, Entropy principle

*Corresponding author: C. Soize, christian.soize@u-pem.fr

Email addresses: christian.soize@u-pem.fr (C. Soize), ghanem@usc.edu (R. Ghanem), csafta@sandia.gov (C. Safta), xhuan@sandia.gov (X. Huan), zvane@sandia.gov (Z. P. Vane), joefelein@sandia.gov (J. Oefelein), glacaze@sandia.gov (G. Lacaze), hnnajm@sandia.gov (H. N. Najm), tang30@llnl.gov (Q. Tang), chen73@llnl.gov (X. Chen)

Notation

$\delta_{kk'}$	=	Kronecker's symbol
E	=	mathematical expectation
$[I_n]$	=	identity matrix in $\mathbb{M}_n$
$\mathbb{M}_n$	=	set of all the square $(n \times n)$ real matrices
$\mathbb{M}_{n,N}$	=	set of all the $(n \times N)$ real matrices
$\mathbb{N}$	=	set of all the integers $0, 1, \ldots$
$\mathbb{R}$	=	set of all the real numbers
$\mathbb{R}^n$	=	Euclidean space of dimension n
$[x]^T$	=	transpose of the real matrix $[x]$
$\|\|[x]\|\|_F$	=	Frobenius norm (Hilbert Schmidt norm) of matrix $[x]$

A lower case letter such as x or η is a real deterministic variable.
A boldface lower case letter such as $\mathbf{x}$ or $\boldsymbol{\eta}$ is a real deterministic vector.
An upper case letter such as X or H is a real random variable.
A boldface upper case letter such as $\mathbf{X}$ or $\mathbf{H}$ is a real random vector.
A lower case letter between brackets such as $[x]$ or $[\eta]$ is a real deterministic matrix.
A boldface upper case letter between brackets such as $[\mathbf{X}]$ or $[\mathbf{H}]$ is a real random matrix.

1. Introduction

Objective and novelty of the paper. This work is a continuation and final installment for previous work [1, 2] devoted to a methodology for probabilistic learning on manifolds. Starting with a specific dataset, the algorithm constructs a statistical model of the data, together with a numerical generator to sample additional statistically consistent realizations. Specifically, the procedure involves three steps consisting of (**1**) identifying the non-Gaussian probability distribution of a random vector for which the solely available information consists of a database made up of independent samples (generated by experiments and/or by numerical simulations), (**2**) delineating an intrinsic manifold in the data and, (**3**) constructing a Markov Chain Monte Carlo (MCMC) generator of realizations on the manifold with the non-Gaussian distribution as its invariant measure. This approach introduces two hyperparameters. The first one, ε, is related to the parameterization of the isotropic-diffusion kernel related to the transition matrix of the diffusion maps used to discover the manifold. The second one, L, is a cut-off threshold for determining the dimension m of the diffusion maps basis used for constructing the MCMC generator. This probabilistic learning algorithm (that depends on ε and L) facilitates the detection and the construction of the intrinsic underlying probability distribution from which the database has been generated. The parameter L is associated with scale separation and a reasonable universal value for it has been deduced from considerations of a large number of datasets from a variety of disciplines. The novelty of the present paper is to propose a selection criterion for ε, resulting in a data-driven model that is algebraically closed.

Role played by statistical learning in computational physics and engineering sciences. Statistical and probabilistic learning methods have been extensively developed (see for instance, [3, 4, 5, 6, 7, 8, 9, 10, 11, 12]) and play an increasingly important role in computational physics and engineering science [13], in particular for design optimization with underlying stochastic operators and constraints using large scale computational model, and more generally in artificial intelligence for extracting information from big data. In this framework, statistical learning

methods have been developed in the form of surrogate models from which approximations of the expensive functions can easily be evaluated [14, 15, 16, 17]. Although Gaussian process models are most commonly used in this context [18], alternative approaches based on Bayesian methods have been proposed [14, 19, 20]. For the evaluations of expensive stochastic functions in the presence of uncertainties, current computational challenges remain significant enough to require some degree of probabilistic approximation [21, 16, 22, 23].

A classical example of the use of the probabilistic learning on manifolds is the following. We consider an expensive large-scale stochastic computational model that discretizes a complex system modeled by a boundary value problem. The vector-valued random response is written as $\mathbf{Q} = \mathbf{f}(\mathbf{W}, \mathbf{U})$ in which $\mathbf{W}$ is a non-Gaussian vector-valued random parameter controlling the system, $\mathbf{U}$ is a non-Gaussian vector-valued random parameter representing uncertainties, and $\mathbf{f}$ is a deterministic nonlinear mapping representing the computational model. For instance $\mathbf{U}$ corresponds to the spatial discretization of a non-Gaussian tensor-valued random field that is a coefficient of a partial differential operator of the boundary value problem on which the computational model is based. The only available information is a given initial dataset (training set) of length N, which is constructed as the set of N points $\{\mathbf{x}_d^j = (\mathbf{q}_d^j, \mathbf{w}_d^j), j = 1, \ldots, N\}$ in which $\mathbf{q}_d^j = \mathbf{f}(\mathbf{w}_d^j, \mathbf{U}(\theta_j))$ are N independent realizations of $\mathbf{Q}$ (calculated with the expensive computational model) where $\mathbf{w}_d^j$ and $\mathbf{U}(\theta_j)$ are N independent realizations of $\mathbf{W}$ and $\mathbf{U}$. Consequently, $\{\mathbf{x}_d^j, j = 1, \ldots, N\}$ are N independent realizations of the non-Gaussian random vector $\mathbf{X} = (\mathbf{Q}, \mathbf{W})$. Knowing only this initial dataset, the objective is, for instance, to construct, for any given $\mathbf{w}$ in its admissible set, an estimate $h^{(N)}(\mathbf{w})$ of $h(\mathbf{w})$ that is defined, for instance, by $h(\mathbf{w}) = E\{H(\mathbf{Q})|\mathbf{W} = \mathbf{w}\}$ in which E is the conditional mathematical expectation given $\mathbf{W} = \mathbf{w}$ and where H is a given deterministic mapping (for instance, $h(\mathbf{w})$ could be the objective function of an optimization problem for which $\mathbf{w}$ would be the design parameter). If each evaluation $\mathbf{q}_d^j$ is expensive in CPU time, then N will be, generally, not sufficiently large for obtaining a good convergence of $h^{(N)}(\mathbf{w})$ towards $h(\mathbf{w})$. The probabilistic learning on manifolds allows for generating $\nu_{\text{sim}} \gg N$ additional independent realizations $\{\mathbf{x}_{\text{ar}}^1, \ldots, \mathbf{x}_{\text{ar}}^{\nu_{\text{sim}}}\}$ of random vector $\mathbf{X}$, without using the expensive computational model, which allows for deducing ν_{sim} additional realizations $(\mathbf{q}_{\text{ar}}^\ell, \mathbf{w}_{\text{ar}}^\ell), \ell = 1, \ldots, \nu_{\text{sim}}\}$ such that $(\mathbf{q}_{\text{ar}}^\ell, \mathbf{w}_{\text{ar}}^\ell) = \mathbf{x}_{\text{ar}}^\ell$. We can then construct a better estimate of the conditional expectation that is required for computing $h^{(N)}(\mathbf{w})$. We are then considering a probabilistic machine learning for the small-data challenge (N small) in computational science.

Brief description of probabilistic learning under consideration and definition of the problem to be solved. As explained above, the authors have proposed a general probabilistic learning on manifold [1] for generating additional realizations of an $\mathbb{R}^n$-valued random variable $\mathbf{X} = (X_1, ..., X_n)$ for which the available information is only made up of a given set of $N \gg n$ points $\{\mathbf{x}^1, \ldots, \mathbf{x}^N\}$ that are N independent realizations in $\mathbb{R}^n$ of random vector $\mathbf{X}$. This method is based (1) on the use of nonparametric statistics for estimating the probability density function $\mathbf{x} \mapsto p_{\mathbf{X}}(\mathbf{x})$ with respect to the Lebesgue measure $d\mathbf{x}$ on $\mathbb{R}^n$ of random vector $\mathbf{X}$ and then for deducing the probability density function $[x] \mapsto p_{[\mathbf{X}]}([x])$ with respect to the volume element $d[x]$ on $\mathbb{M}_{n,N}$ of the random matrix $[\mathbf{X}] = [\mathbf{X}^1, \ldots, \mathbf{X}^N]$ with values in $\mathbb{M}_{n,N}$ in which $\mathbf{X}^1, \ldots, \mathbf{X}^N$ are N independent copies of $\mathbf{X}$, (2) on the construction of nonlinear Itô stochastic differential equation (ISDE) on $\mathbb{M}_{n,N}$, formulated for a dissipative Hamiltonian dynamical system for which $p_{[\mathbf{X}]}([x])\, d[x]$ is the invariant measure, (3) on the use of diffusion maps for discovering the geometrical structure of the given set of points, and (4) on the construction of a reduced ISDE on $\mathbb{M}_{n,m}$ obtained by projecting the original ISDE on a subspace of dimension $m \ll N$ spanned by

a subset of the diffusion-maps basis and allowing additional realizations of $\mathbf{X}$ to be generated. This approach introduces two hyperparameters. The first one is the isotropic-diffusion kernel hyperparameter ε allowing the transition matrix of the diffusion-maps approach to be constructed from the given set of points. The second one is dimension m of the projection. Concerning hyperparameter ε related to the isotropic diffusion kernel, the authors did not find any proposed robust method in the open literature for estimating it on the basis of a quantitative criterion independent of the application under consideration. In addition, in the framework of the probabilistic learning on manifolds proposed in [1], dimension m is deduced from ε once a scale separation threshold, L, has been set. In this paper, we propose a selection model of ε based on the use of an entropy principle using only the given set of points. The algebraic basis constructed through the diffusion manifold approach is critical to the projected Itô equation developed in the proposed probabilistic learning method. Other manifold detection procedures, such as kernel PCA [3], that are endowed with such algebraic constructs could be used as substitutes for the manifold detection and characterization step. The probabilistic learning method proposed in [1] has been extended to polynomial chaos representation of databases on manifolds [2] and to probabilistic nonconvex constrained optimization with a fixed number of function evaluations [24]. Recently, the robustness of this approach has been tested with success on different problems such as the optimal well-placement [25, 26], the prediction of maximum daily precipitation with data generated from large scale climate models, the Continental United States (CONUS) Regionally-Refined Model (RRM) [27] of Energy Exascale Earth System Model (E3SM) V0 [28], and the enhancing of the predictability of a two-dimensional (2D) computational model for a scramjet [29].

Organization of the paper. The objectives and the organization of the paper are as follows. Despite the fact that the reader can find the details of the considered probabilistic learning algorithm on manifolds in [1], and taking into account that the objective of the paper is to propose a selection model for its hyperparameters, ε and m, it seems reasonable to start by summarizing this algorithm. Consequently, Section 2 presents such a summary of the probabilistic learning on manifolds that underscores the role played by hyperparameters ε and m. Section 3 is devoted to a selection model for calculating an optimal value ε^{opt} of hyperparameter ε using an entropy principle and a method for computing an adapted dimension $m_L(\varepsilon)$ of hyperparameter $m(\varepsilon)$ as a function of ε and consequently, deducing the optimal value m^{opt} of $m(\varepsilon)$ associated with ε^{opt}. Section 4 deals with a reanalysis of Application 2 presented in [1] (random 3D-data around a helix) using the proposed selection model. In Section 5, a validation of the proposed selection model of the hyperparameters is presented for several databases: Scramjet-d8 and ScramJet-d16 that correspond to simulations of a ScramJet performed by Sandia National Laboratories in Livermore with two large scale complex computational models [29], and Climate-LS-21-34 and Climate-LS-21-50 that correspond to the maxima of daily precipitations, which have been predicted by Lawrence Livermore National Laboratory with a climate computational model [30]. Section 6 presents an analysis of the results obtained.

2. Summary of the probabilistic learning on manifolds

The probabilistic learning on manifold proposed in [1] uses only a given set of N points $\{\mathbf{x}_d^1, \ldots, \mathbf{x}_d^N\}$ in $\mathbb{R}^n$, which are assumed to be N independent realizations of a random vector $\mathbf{X}$ with values in $\mathbb{R}^n$. The probability distribution of $\mathbf{X}$ is unknown and is assumed to be concentrated in a neighborhood of a subset of $\mathbb{R}^n$ (a manifold) that is also unknown and that has to

be discovered. We introduce the matrix $[x_d] = [\mathbf{x}_d^1 \dots \mathbf{x}_d^N] \in \mathbb{M}_{n,N}$ that is one realization of the random matrix $[\mathbf{X}] = [\mathbf{X}^1, \dots, \mathbf{X}^N]$ with values in $\mathbb{M}_{n,N}$ in which $\mathbf{X}^1, \dots, \mathbf{X}^N$ are N independent copies of $\mathbf{X}$. For the dataset that will be used in this paper, vector $\mathbf{X}$ will consist of the parameters of a computational model as well as the quantities of interest that are obtained as output of the computational model. The objective of the probabilistic learning on manifold is to construct a probabilistic model using only the given dataset represented by $[x_d]$. This model is then used to generate additional independent realizations $\{\mathbf{x}_{\text{ar}}^1, \dots, \mathbf{x}_{\text{ar}}^{\nu_{\text{sim}}}\}$ in $\mathbb{R}^n$ of random vector $\mathbf{X}$. The proposed method preserves the concentration of the additional realizations around the manifold. For the applications described subsequently in the paper, a very large number, $\nu_{\text{sim}} \gg N$, of additional realizations can then be generated for the quantities of interest, which then allow for estimating the probability density functions of various quantities of interest (QoI), including extreme values statistics. The main steps of the probabilistic learning algorithm on manifold are summarized in Section 2.1. Section 2.2 deals with the important issue concerning the convergence analysis with respect to hyperparameter m for a given value of ε and the non-adaptation of such a convergence analysis to the objective of the probabilistic learning method. Section 2.3 deals with the mathematical formulation of an efficient method for calculating dimension m as a function of ε.

2.1. Probabilistic learning algorithm on manifold

The 8 steps of the probabilistic learning algorithm used in this paper are summarized next.

1) The initial given dataset is made up of unscaled data represented by matrix $[x_d^{uns}]$ in $\mathbb{M}_{n,N}$. The matrix $[x_d]$ in $\mathbb{M}_{n,N}$ of the scaled given dataset (simply called the given dataset) is constructed such that $[x_d]_{kj} = ([x_d^{uns}]_{kj} - \min_{j'}[x_d^{uns}]_{kj'})/(\max_{j'}[x_d^{uns}]_{kj'} - \min_{j'}[x_d^{uns}]_{kj'})$ for all $k = 1, \dots, n$ and $j = 1, \dots, N$. This step is generally required to avoid numerical problems in the second step introduced below.
2) Let $[c] \in \mathbb{M}_n$ be the empirical estimate of the covariance matrix of $\mathbf{X}$. We consider the eigenvalue problem $[c]\,\boldsymbol{\varphi}^k = \mu_k\,\boldsymbol{\varphi}^k$. Let $\nu \leq n$ be the number of positive eigenvalues $\{\mu_k\}_{k=1}^{\nu}$ with $0 < \mu_1 \leq \mu_2 \leq \dots \leq \mu_\nu$ and let $[\varphi]$ be the $(n \times \nu)$ matrix such $[\varphi]^T\,[\varphi] = [I_\nu]$, whose columns are the associated orthonormal eigenvectors $\boldsymbol{\varphi}^1, \dots, \boldsymbol{\varphi}^\nu$. A principal component analysis $[\mathbf{X}] = [\underline{x}] + [\varphi]\,[\mu]^{1/2}\,[\mathbf{H}]$ of random matrix $[\mathbf{X}]$ is carried out in order to normalize/reduce the dataset in which $[\underline{x}] \in \mathbb{M}_{n,N}$ is the empirical estimate of the mean value of $[\mathbf{X}]$ and where $[\mu]$ is the positive diagonal $(\nu \times \nu)$ real matrix such that $[\mu]_{kk'} = \delta_{kk'}\mu_k$. The columns $\mathbf{H}^1, \dots, \mathbf{H}^N$ of random matrix $[\mathbf{H}]$ with values in $\mathbb{M}_{\nu,N}$ are independent random vectors with values in $\mathbb{R}^\nu$. This normalized/reduced representation allows for computing a new normalized dataset of N points $\{\boldsymbol{\eta}_d^1, \dots, \boldsymbol{\eta}_d^N\}$ in $\mathbb{R}^\nu$, represented by the matrix $[\eta_d] = [\boldsymbol{\eta}_d^1 \dots \boldsymbol{\eta}_d^N]$ in $\mathbb{M}_{\nu,N}$ that is computed by $[\eta_d] = [\mu]^{-1/2}[\varphi]^T\,([x_d] - [\underline{x}])$. If ν is chosen as the rank of matrix $[c]$, then the constructed representation of $[\mathbf{X}]$ induces no error and corresponds to a pure normalization. If ν is chosen less that the rank of $[c]$ in order to construct a reduced-order representation of $[\mathbf{X}]$, then ν can be selected for obtaining a given tolerance of the relative mean-square error defined by $\text{error}_{pca}(\nu) = \|\,[x_d] - [x_d^{(\nu)}]\,\|_F / \|\,[x_d]\,\|_F$ where $[x_d^{(\nu)}] = [\underline{x}] + [\varphi]\,[\mu]^{1/2}\,[\eta_d]$.
3) A modification [31] of the classical multidimensional Gaussian kernel-density estimation method [32, 33] is then used to construct an estimate of the probability density function $[\eta] \mapsto p_{[\mathbf{H}]}([\eta])$ with respect to the volume element $d[\eta]$ on $\mathbb{M}_{\nu,N}$ of random matrix $[\mathbf{H}]$.
4) An MCMC generator for random matrix $[\mathbf{H}]$ is constructed using the approach proposed in [34, 31] belonging to the class of Hamiltonian Monte Carlo methods [34, 35, 36], which

is an MCMC algorithm [37]. The realizations of random matrix $[\mathbf{H}]$ could be obtained by solving a $(\nu \times N)$ matrix-valued ISDE that corresponds to a stochastic nonlinear dissipative Hamiltonian dynamical system, for which $p_{[\mathbf{H}]}([\eta])\, d[\eta]$ is the unique invariant measure (in fact, we will not need to generate realizations of $[\mathbf{H}]$ using this ISDE).

5) The diffusion-maps approach [38, 39] is then used to discover and characterize the geometrical structure of the normalized dataset $[\eta_d]$ thus defining a manifold embedded in $\mathbb{R}^N$. The transition matrix of a Markov chain relative to the geometrical structure of the given normalized dataset $[\eta_d]$ is constructed thanks to the introduction of the isotropic diffusion kernel, $k_\varepsilon(\boldsymbol{\eta}, \boldsymbol{\eta}') = \exp(-\frac{1}{4\varepsilon}\|\boldsymbol{\eta} - \boldsymbol{\eta}'\|^2)$, defined on $\mathbb{R}^\nu \times \mathbb{R}^\nu$ in which $\varepsilon > 0$ is a hyperparameter of the model. Let $[K]$ be the symmetric matrix in $\mathbb{M}_N$ with positive entries such that $[K]_{ij} = k_\varepsilon(\boldsymbol{\eta}_d^i, \boldsymbol{\eta}_d^j)$ for i and j in $\{1, \ldots, N\}$. Let $[b]$ be the positive-definite diagonal real matrix in $\mathbb{M}_N$ such that $[b]_{ij} = \delta_{ij} \sum_{j'=1}^N [K]_{ij'}$ and let $[\mathbb{P}]$ be the matrix in $\mathbb{M}_N$ such that $[\mathbb{P}] = [b]^{-1}\,[K]$. Matrix $[\mathbb{P}]$ that has positive entries satisfying $\sum_{j=1}^N [\mathbb{P}]_{ij} = 1$ for all $i = 1, \ldots, N$, can be viewed as the transition matrix of a Markov chain that yields the probability of transition in one step. We consider the first $\widehat{m}$ largest eigenvalues of the generalized eigenvalue problem $[K]\,[\psi] = [b]\,[\psi]\,[\Lambda]$ such that $[\psi]^T\,[b]\,[\psi] = [I_{\widehat{m}}]$, where $[I_{\widehat{m}}]$ is the identity matrix in $\mathbb{M}_{\widehat{m}}$ and where $[\psi]^T\,[K]\,[\psi] = [\Lambda]$. The diagonal matrix $[\Lambda]$ in $\mathbb{M}_{\widehat{m}}$ is made up of the eigenvalues $\lambda_1, \ldots, \lambda_{\widehat{m}}$ that are such that $1 = \lambda_1 > \lambda_2 \geq \ldots \geq \lambda_{\widehat{m}}$. The corresponding eigenvectors $\boldsymbol{\psi}^1, \ldots, \boldsymbol{\psi}^{\widehat{m}}$ in $\mathbb{R}^N$ are such that $[\psi] = [\boldsymbol{\psi}^1 \ldots, \boldsymbol{\psi}^{\widehat{m}}] \in \mathbb{M}_N$. The diffusion-maps basis of dimension $\widehat{m}$ is then defined by the $\widehat{m}$ vectors $\mathbf{g}^1, \ldots, \mathbf{g}^{\widehat{m}}$ in $\mathbb{R}^N$ such that $\mathbf{g}^\alpha = \lambda_\alpha \boldsymbol{\psi}^\alpha$. We introduce the $(N \times m)$ matrix $[g] = [\mathbf{g}^2 \ldots \mathbf{g}^{\widehat{m}}]$ with $m = \widehat{m} - 1$, which will be used below for performing the projection of random matrix $[\mathbf{H}]$. In general, m can be selected such that $m \ll N$ (see Section 2.2). Note that $\mathbf{g}^1$ is always a vector whose components are all equal. Since the random matrix $[\mathbf{H}]$ is centered, this vector can be removed from matrix $[g]$ used for performing the projection of $[\mathbf{H}]$. Finally, since eigenvalues $\lambda_2, \ldots, \lambda_{\widehat{m}}$ and matrix $[g]$ depend on ε, we will rewrite them as $\lambda_2(\varepsilon), \ldots, \lambda_{\widehat{m}}(\varepsilon)$ and $[g^{\varepsilon,m}]$.

6) As proposed in [1], a reduced-order representation $[\mathbf{H}^{\varepsilon,m}]$ of $[\mathbf{H}]$ is introduced such that $[\mathbf{H}^{\varepsilon,m}] = [\mathbf{Z}^{\varepsilon,m}]\,[g^{\varepsilon,m}]^T$ is constructed on the manifold in which $[\mathbf{Z}^{\varepsilon,m}]$ is a random matrix with values in $\mathbb{M}_{\nu,m}$ for which $m \ll N$ and a $\mathbb{M}_{\nu,m}$-valued reduced-ISDE is obtained by projecting the $\mathbb{M}_{\nu,N}$-valued ISDE onto the diffusion manifold by using the reduced-order basis represented by matrix $[g^{\varepsilon,m}]^T$. It should be noted that such a projection corresponds to a reduction of the dataset dimension and not to a reduction of the physical components of random vector $\mathbf{H}^\ell$ (this latter reduction already results from a statistical reduction introduced in step 2). Such a projection preserves the concentration of the generated realizations around the manifold.

7) The constructed reduced ISDE is then used for generating n_{MC} additional realizations $[z_{\text{ar}}^1], \ldots, [z_{\text{ar}}^{n_{\text{MC}}}]$ of random matrix $[\mathbf{Z}^{\varepsilon,m}]$, and therefore, for deducing the additional realizations $[\eta_{\text{ar}}^1], \ldots, [\eta_{\text{ar}}^{n_{\text{MC}}}]$ of random matrix $[\mathbf{H}^{\varepsilon,m}]$. Using step 2 yields n_{MC} additional realizations $[x_{\text{ar}}^1], \ldots, [x_{\text{ar}}^{n_{\text{MC}}}]$ of $[\mathbf{X}^{\varepsilon,m}]$ such that $[x_{\text{ar}}^\ell] = [\underline{x}] + [\varphi]\,[\mu]^{1/2}\,[\eta_{\text{ar}}^\ell]$. The algorithm for generating these additional realizations is given in Appendix A.

8) Reshaping these n_{MC} matrices yields the $\nu_{\text{sim}} = N \times n_{\text{MC}} \gg N$ additional realizations $\{\mathbf{x}_{\text{ar}}^1, \ldots, \mathbf{x}_{\text{ar}}^{\nu_{\text{sim}}}\}$ in $\mathbb{R}^n$ of random vector $\mathbf{X}^{\varepsilon,m}$.

2.2. Remark concerning the convergence analysis with respect to hyperparameter m for a given value of ε and its non-adaptation to the objective of the probabilistic learning method

For a given application for which the value of hyperparameter ε is fixed, the corresponding probability density function $[x] \mapsto p_{[\mathbf{X}^{\varepsilon,m}]}([x];\varepsilon,m)$, with respect to $d[x]$, of random matrix $[\mathbf{X}^{\varepsilon,m}]$ corresponds to a better construction than the one used for the probability density function $[x] \mapsto p_{[\mathbf{X}]}([xd])$, with respect to $d[x]$, of random matrix $[\mathbf{X}]$, which would be estimated by using only the Gaussian kernel-density estimation method from the given dataset represented by $[x_d]$, because $p_{[\mathbf{X}^{\varepsilon,m}]}([x];\varepsilon,m)$ also uses a second source of information relative to the geometrical structure of the given dataset. An interpretation of the difference between $p_{[\mathbf{X}^{\varepsilon,m}]}([x];\varepsilon,m)$ and $p_{[\mathbf{X}]}([xd])$ is necessary and is presented next.

It should be noted that for $\widehat{m} = N$, the family of vectors $\{\mathbf{g}^1, \mathbf{g}^2, \ldots, \mathbf{g}^N\}$ constitutes a vector basis of $\mathbb{R}^N$. An estimation of $\widehat{m}$ could be constructed with a convergence analysis aimed at reducing the relative mean-square error defined by $\text{error}_{proj}(\widehat{m}) = \| [x_d] - [x_d^{(\nu,\widehat{m})}] \|_F / \| [x_d] \|_F$ in which $[x_d^{(\nu,\widehat{m})}] = [\underline{x}] + [\varphi]\,[\mu]^{1/2}\,[z_d]\,[g]^T$ with $[z_d] = [y_d]\,[g]\,([g]^T[g])^{-1}$. As this error goes to zero, $\widehat{m}$ goes to N and the

probability measure $p_{[\mathbf{X}^{\varepsilon,m}]}([x];\varepsilon,N)\,d[x]$ approaches the probability measure $p_{[\mathbf{X}]}([x])\,d[x]$. In such a limit, the concentration of data around the manifold would not be leveraged, and there would be no gain achieved using the proposed probabilistic learning algorithm [1]. Thus, the L_2 convergence analysis described above is not consistent with the objectives of the proposed probabilistic learning methodology. As already explained, the main objective of this probabilistic learning method is to take into account the geometrical structure of the given dataset in order to enrich the construction of the probability measure $p_{[\mathbf{X}]}([x])\,d[x]$ substituting it by $p_{[\mathbf{X}^{\varepsilon,m}]}([x];\varepsilon,m)\,d[x]$ that preserves the concentration of the additional realizations computed with steps 7 and 8 of the algorithm presented in Section 2.1. Thus, rather than pursuing a convergence analysis by increasing m, an optimal value of m is selected for each value of ε. In several of the numerical examples presented in Section 5, probability density estimates have also been constructed for extreme values statistics of random QoIs. The foregoing discussion has the implication that the convergence of these extreme values statistics with respect to m should not be of concern, as m is selected with the criterion of sufficient scale separation. The model is completely specified by selecting numerical values for m and ε. Consequently, these statistics correspond to the extreme values statistics of the QoI that are defined relative to this selected model.

2.3. Method for calculating an adapted dimension $m_L(\varepsilon)$ as a function of ε

As explained in Section 1, for a given application, the value of hyperparameter ε is currently selected arbitrarily, following a trial and error procedure with no quantitative criteria.

A plausible framework for separating scales in a given dataset would first involve evaluating, for different values of ε, a graph of the eigenvalues $\alpha \mapsto \lambda_\alpha(\varepsilon)$ for $\alpha \geq 0$, similar to the ones displayed in Fig. 1. It is clear that different values of ε result in different decays in the eigenvalues, and a different value of an optimal m corresponding to the rank of the eigenvalue at which significant drop in the spectrum is observed. It is also clear from these graphs that the lagest eigenvalue is always equal to 1 and its eigenvector is constant, an artifact of the normalization of the Laplacian of the data (ie the matrix obtained from discretizing the diffusion kernel). In all subsequent calculations, this largest "unit" eigenvalue is therefore ignored when comparing spectral content of different models. A method for identifying ε objectively is presented in Section 3. In the rest of this section, we present a procedure for evaluating the optimal value of m

for a fixed value of ε. The model proposed for calculating the adapted value $\widehat{m}_L(\varepsilon)$ of $\widehat{m}(\varepsilon)$ as a function of ε can be written as,

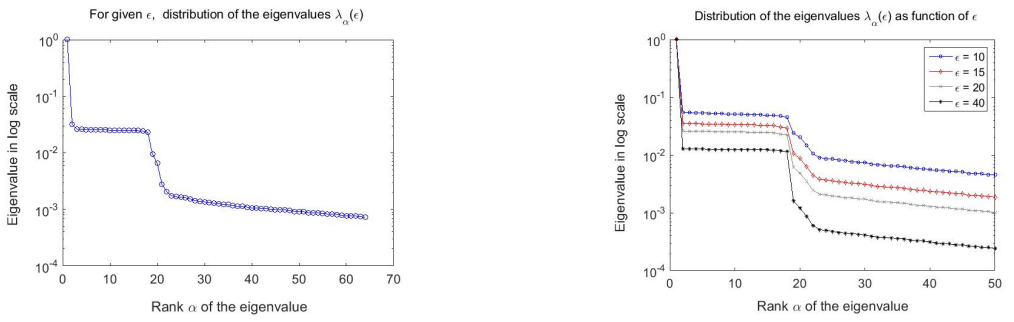
$$\widehat{m}_L(\varepsilon) = -1 + \arg\min_{\alpha \,|\, \alpha \geq 3} \left\{ \frac{\lambda_\alpha(\varepsilon)}{\lambda_2(\varepsilon)} < L \right\} , \tag{1}$$

and then the corresponding adapted dimension $m_L(\varepsilon)$ of dimension m is such that

$$m_L(\varepsilon) = \widehat{m}_L(\varepsilon) - 1 \, . \tag{2}$$

Parameter L is independent of ε and is chosen for separating the existing scales in any given dataset. Following our proposed procedure for estimating ε, L will be the only remaining free parameter in our data-driven model, although its significance as a scale separation threshold is obvious form context. All the numerical applications performed by the authors for dataset coming from different fields show that $L = 0.1$ is, generally, a good value and this choice corresponds to one order of magnitude for a scale separation. The analyses that will be performed in the future for other datasets will allow this choice of the value of L to be confirmed or improved. For the case illustrated in Fig. 1(a), we have $\widehat{m}_L(\varepsilon) = 20$ and $m_L(\varepsilon) = 19$ because $\lambda_{21}(\varepsilon)/\lambda_2(\varepsilon) = 0.0028/0.0316 < 0.1$ while $\lambda_{20}(\varepsilon)/\lambda_2(\varepsilon) = 0.0065/0.0316 > 0.1$. As an illustration, Fig. 1(b) displays the distributions of eigenvalues $\{\lambda_\alpha(\varepsilon)\}_\alpha$ for $\varepsilon \in \{10, 15, 20, 40\}$ for a given database (Scramjet-d16 database) and allows for highlighting the construction defined by Eq. (1).

The criterion defined by Eq. (1), is based on the idea that, for a fixed value of ε, the smallest dimension $m(\epsilon)$ has to be chosen for obtaining a scale separation and for preserving the concentration of the additional realizations generated by the probabilistic learning method that has been proposed [1]. In this paper, we propose a selection model for calculating an optimal value ε^{opt} of ε and then, deducing the corresponding optimal value $m^{\text{opt}} = m_L(\varepsilon^{\text{opt}})$ of dimension m. This model is based on the use of an entropy principle and is presented in Section 3.



(a) (b)

Figure 1: (a): choice of ε yielding a strong decreasing of the eigenvalues $\lambda_\alpha(\varepsilon)$ for which the ranks are greater than 18. The graph shows that $\lambda_2(\varepsilon) = 0.0316$, $\lambda_{18}(\varepsilon) = 0.0231$, $\lambda_{20}(\varepsilon) = 0.0065$, and $\lambda_{21}(\varepsilon) = 0.0028$. In the present case, for the criterion defined by Eq. (1), we have $\widehat{m}_L(\varepsilon) = 20$ and $m_L(\varepsilon) = 19$. (b): illustration of the distributions of eigenvalues $\{\lambda_\alpha(\varepsilon)\}_\alpha$ for $\varepsilon = 10$ (blue circle), $= 15$ (red diamond), $= 20$ (green cross), and $= 40$ (black star) for a given database (Scramjet-d16 database).

3. Selection model for calculating an optimal value ε^{opt} of hyperparameter ε

For given ε and m, let $[\eta] \mapsto p_{[\mathbf{H}^{\varepsilon,m}]}([\eta]\,;\varepsilon,m)$ be the probability density function of random matrix $[\mathbf{H}^{\varepsilon,m}]$ with respect to the volume element $d[\eta]$ on $\mathbb{M}_{\nu,N}$, for which the additional realizations $[\eta_{\text{ar}}^1],\ldots,[\eta_{\text{ar}}^{n_{\text{MC}}}]$ are computed as explained in step 7 of Section 2.1. Note that ν and N of random matrix $[\mathbf{H}^{\varepsilon,m}]$ are independent of ε and m.

3.1. Principles for selecting the optimal value of hyperparameter ε

The principles that are proposed for selecting the optimal value ε^{opt} of hyperparameter ε and the corresponding optimal value m^{opt} of hyperparameter m are the following:

1) An admissible finite set $\mathcal{A} = \{\varepsilon_1 < \varepsilon_2 < \ldots < \varepsilon_{n_\varepsilon}\}$ of n_ε ordered values of ε is defined. The optimal value ε^{opt} of ε will be searched for in $\mathcal{A}$. Consequently, for a given database, set $\mathcal{A}$ must be chosen to contain a sufficiently large number of possible values of ε, but the selection model that is proposed does not directly depend on this choice.
2) For each value of ε in $\mathcal{A}$, the adapted value $\widehat{m}_L(\varepsilon)$ of $\widehat{m}(\varepsilon)$ is computed with Eq. (1) for the reasons given in Section 2.3. The corresponding adapted dimension $m_L(\varepsilon)$ of hyperparameter m is then computed using Eq. (2).
3) The mapping $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ from $\mathcal{A}$ into $\mathbb{N}$ being known, we define the subset $\mathcal{E}_0$ of $\mathcal{A}$, made up of all the values of ε that minimize $m_L(\varepsilon)$ on $\mathcal{A}$,

$$\mathcal{E}_0 = \{\varepsilon \in \mathcal{A} \mid \varepsilon = \arg\min_{\varepsilon' \in \mathcal{A}} \widehat{m}_L(\varepsilon')\}\,, \tag{3}$$

   In general, the number of elements in set $\mathcal{E}_0$ is greater than 1 (but is less or equal to n_ε). This is one of the reasons why ε^{opt} cannot simply be constructed as the value of ε that minimizes $\widehat{m}_L(\varepsilon)$.
4) We now construct the subset $\mathcal{E} \subset \mathcal{A}$ of the values of ε such that

$$\mathcal{E} = \left\{\varepsilon \in \mathcal{A} \mid \widehat{m}_L^{\min} \leq \widehat{m}_L(\varepsilon) \leq \widehat{m}_L^{\max}\right\}\,, \tag{4}$$

   in which the lower bound $\widehat{m}_L^{\min}$ is the minimum value of $\widehat{m}_L(\varepsilon)$ for $\varepsilon \in \mathcal{A}$,

$$\widehat{m}_L^{\min} = \min_{\varepsilon \in \mathcal{A}} \widehat{m}_L(\varepsilon)\,, \tag{5}$$

   and where $\widehat{m}_L^{\max}$ is the upper bound defined by

$$\widehat{m}_L^{\max} = \lfloor (1+\chi)\, \widehat{m}_L^{\min} \rfloor\,, \tag{6}$$

   with $\lfloor r \rfloor$ denotes the floor integer part of r (for instance, $\lfloor 2.3 \rfloor = 2$), and where χ is a small real number such that $0 \leq \chi < 1$. It can be seen that

$$\mathcal{E}_0 \subseteqq \mathcal{E} \subseteqq \mathcal{A}\,. \tag{7}$$

   The optimal value ε^{opt} of ε will be searched into $\mathcal{E}$ and not in $\mathcal{E}_0$ in order to construct a subset of $\mathcal{A}$, the subset $\mathcal{E}$, which is more robust than $\mathcal{E}_0$ with respect to the small fluctuations effects that can occur with Eq. (1). Based on our experience of the analysis of data sets from a range of applications, for the value $L = 0.1$ recommended in Section 2.3, the value $\chi = 0.2$ is a good associated value. It should be noted that χ is not a hyperparameter of the probabilistic learning approach as is parameter ε whose value depends on

the database and for which a procedure is proposed for calculating it, but χ is a parameter whose value is independent of the database and for which a fixed value is proposed resulting from the analyses of several databases (may be this value of χ could be improved with future experimentations).

5) Subset $\mathcal{E}$ being known, the optimal value ε^{opt} is sought as the value of ε in $\mathcal{E}$, which maximizes the uncertainty of the family of random matrices $\{[\mathbf{H}^{\varepsilon,m_L(\varepsilon)}], \varepsilon \in \mathcal{E}\}$. This is equivalent [40, 41, 42, 13, 43] to maximizing the Shannon entropy [44] with respect to ε for the probability density function of random matrix $[\mathbf{H}^{\varepsilon,m_L(\varepsilon)}]$. This entropy $S(\varepsilon)$ is then written as

$$S(\varepsilon) = -\int_{\mathbb{M}_{\nu,N}} p([\eta]\,;\varepsilon)\, \log(p([\eta]\,;\varepsilon))\, d[\eta]\,, \tag{8}$$

in which we have introduced the following simplified notation,

$$p([\eta];\varepsilon) = p_{[\mathbf{H}^{\varepsilon,m_L(\varepsilon)}]}([\eta]\,;\varepsilon, m_L(\varepsilon))\,. \tag{9}$$

Consequently, the use of the maximum entropy principle allows the optimal value ε^{opt} of ε to be selected in $\mathcal{E}$, solving the optimization problem,

$$\varepsilon^{\text{opt}} = \max_{\varepsilon\in\mathcal{E}} S(\varepsilon)\,. \tag{10}$$

6) The corresponding optimal value m^{opt} of hyperparameter m is then written as,

$$m^{\text{opt}} = m_L(\varepsilon^{\text{opt}})\,. \tag{11}$$

Remarks.

1) An alternative selection model would consist of replacing principles 4 and 5 above by one consisting in maximizing the entropy over $\mathcal{A}$, that is to say $\varepsilon^{\text{opt}} = \max_{\varepsilon\in\mathcal{A}} S(\varepsilon)$. However, such a principle would favor the selection of epsilon with respect to the criterion of minimizing m^{opt} subject to scale separation.
2) The number of random variables in random matrix $[\mathbf{Z}^{\varepsilon,m}]$ is $\nu \times m$. Consequently, for ε fixed, random matrix $[\mathbf{H}^{\varepsilon,m}] = [\mathbf{Z}^{\varepsilon,m}]\,[g^{\varepsilon,m}]^T$, whose dimension $(\nu \times N)$ is independent of m, depends on $\nu \times m$ random variables. Consequently, the entropy $S(\varepsilon)$ should "grow on average" with $m = m_L(\varepsilon)$. This phenomenon has been observed for all the databases analyzed (see Figs. 5, 7, 9, and 11) except for the simple database associated with a 3D helix (see Fig. 3) for which that is not true for the two small values 0.5 and 1 of ε, but what is true for $\varepsilon \geq 2$. This growth of entropy with m is consistent with the observation that as m increases, less structure is delineated in the data (less localization), resulting in greater uncertainty about mechanisms underlying the fluctuations.

3.2. Computational aspects

The computational aspects concern the numerical calculation of $S(\varepsilon)$ for ε given in $\mathcal{A}$, the solution of the optimization problem, and the convergence analysis with respect to the number n_{MC} of the number of realizations.

3.2.1. Numerical calculation of the entropy

The entropy defined by Eq. (4) is classically rewritten as,

$$S(\varepsilon) = -E\{\log(p([\mathbf{H}^{\varepsilon,m_L(\varepsilon)}]\,;\varepsilon))\}\,, \tag{12}$$

For ε fixed in $\mathcal{A}$ and for $m = m_L(\varepsilon)$, the additional realizations of random matrix $[\mathbf{H}^{\varepsilon,m}]$, which are computed as explained in step 7 of Section 2.1, are denoted by $[\eta^1_{\text{ar}}(\varepsilon)], \ldots, [\eta^{n_{\text{MC}}}_{\text{ar}}(\varepsilon)]$. For n_{MC} sufficiently large, the classical estimate of the right-hand-side of Eq. (12) is written as

$$S(\varepsilon) \simeq -\frac{1}{n_{\text{MC}}} \sum_{\ell=1}^{n_{\text{MC}}} \log(p^{(n_{\text{MC}})}([\eta^\ell_{\text{ar}}(\varepsilon)]\,;\varepsilon))\,, \tag{13}$$

in which the estimation, $p^{(n_{\text{MC}})}([\eta]\,;\varepsilon)$, of $p([\eta];\varepsilon)$ is performed using the modification [31] of the classical multidimensional Gaussian kernel-density estimation method [32, 33] and using additional realizations $[\eta^1_{\text{ar}}(\varepsilon)], \ldots, [\eta^{n_{\text{MC}}}_{\text{ar}}(\varepsilon)]$.

3.2.2. Solution of the optimization problem.

The mapping $\varepsilon \mapsto S(\varepsilon)$ is generally not convex on $\mathcal{E}$. Since $\mathcal{E}$ is a finite subset of $\mathcal{A}$, the optimization problem defined by Eq. (10) is solved using an exhaustive search over $\mathcal{E}$.

3.2.3. Convergence with respect to the number of additional realizations

As the entropy is estimated using n_{MC} samples, the optimal values ε^{opt} and $\widehat{m}^{\text{opt}}$ (or $m^{\text{opt}} = \widehat{m}^{\text{opt}} - 1$) depend of the number n_{MC} of additional realizations generated with step 7 of Section 2.1. The convergence of the optimal values can be analyzed studying the simple convergence in $\mathbb{R}$ of the sequences of real numbers $\{\varepsilon^{\text{opt}}(n_{\text{MC}})\}_{n_{\text{MC}}}$ and $\{\widehat{m}^{\text{opt}}(n_{\text{MC}})\}_{n_{\text{MC}}}$.

4. Analysis of a simple example: 3D-data around a helix (spiral)

For this simple example (similar to application 2 presented in [1]), the initial given dataset is constituted of $N = 400$ independent realizations of random vector $\mathbf{X} = (X_1, X_2, X_3)$ in dimension $n = 3$ and is represented by matrix $[x^d]$ in $\mathbb{M}_{n,N}$. These given data points are concentrated in the neighborhood of a helix as shown in Fig. 2. The principal component analysis (step 2) yields $\nu = 3$ for which $\text{error}_{pca}(\nu) = 1.46 \times 10^{-16}$. The additional realizations are computed (step 7) with $n_{\text{MC}} = 100$ yielding $\nu_{\text{sim}} = 40{,}000$. The admissible finite set $\mathcal{A}$ of the values of ε is chosen to be $\{0.5, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. The graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ and $\varepsilon \mapsto S(\varepsilon)$ defined on $\mathcal{A}$ are displayed in Fig. 3 (left and right). We have $\widehat{m}^{\min}_L = \widehat{m}^{\max}_L = 4$ and $\mathcal{E}_0 = \mathcal{E} = \{5, 6, 7, 8, 9, 10\} \subset \mathcal{A}$. The entropy-based selection yields $\varepsilon^{\text{opt}} = 6$ and consequently, $\widehat{m}^{\text{opt}} = 4$, $m^{\text{opt}} = 3$. The relative mean-square error is $\text{error}_{proj}(\widehat{m}^{\text{opt}}) = 3.41 \times 10^{-3}$. For $\varepsilon = \varepsilon^{\text{opt}}$, Fig. 4(a) shows function $\alpha \mapsto \lambda_\alpha$ for which $\lambda_2 = 8.25 \times 10^{-2}$, $\lambda_4 = 8.06 \times 10^{-2}$, and $\lambda_5 = 6.88 \times 10^{-3}$. As an example of additional realizations generated by the probabilistic learning algorithm, Fig. 4(b) displays 40,000 additional realizations of $\mathbf{X} = (X_1, X_2, X_3)$. These realizations stay concentrated in the neighborhood of the helix (the concentration is effectively preserved).

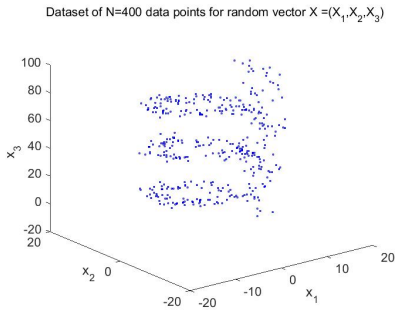
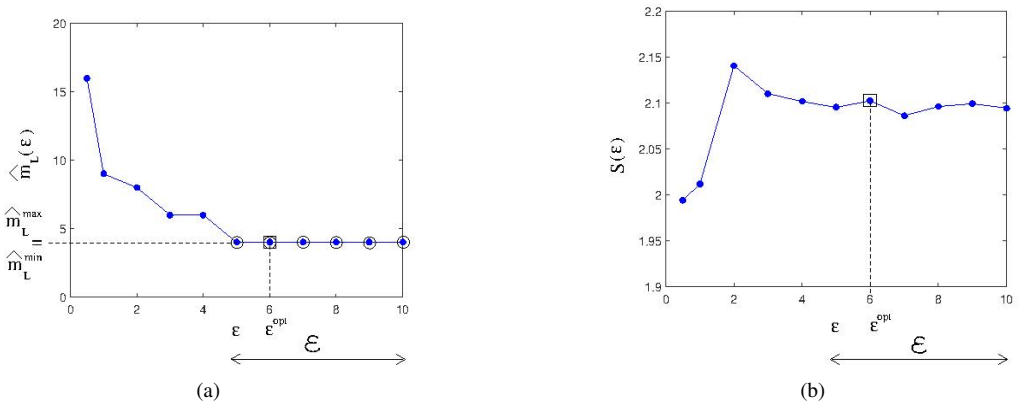


Figure 2: Helix database: initial given dataset constituted of $N = 400$ data points for random vector $\mathbf{X} = (X_1, X_2, X_3)$ concentrated in the neighborhood of a helix.



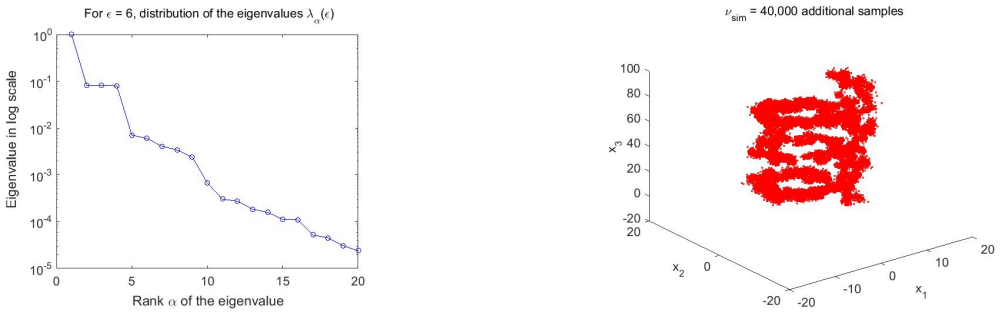
(a) (b)

Figure 3: Helix database: graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ (left figure) and $\varepsilon \mapsto S(\varepsilon)$ (right figure), defined on $\mathcal{A}$. On the two figures, (i) $\mathcal{E} \subset \mathcal{A}$. is the set of the values of ε such that $\widehat{m}_L^{\min} \leq \widehat{m}_L(\varepsilon) \leq \widehat{m}_L^{\max}$, (ii) the values of $\widehat{m}_L(\varepsilon)$ surrounded by circle symbols correspond to the values of ε that minimize $m_L(\varepsilon)$ and that define the set $\mathcal{E}_0 = \mathcal{E}$, and (iii) the square symbol localizes the optimal value ε^{opt} of ε.

5. Applications to several databases

In this section, we present the application of the selection model proposed for several databases: Scramjet-d8 and ScramJet-d16 that correspond to simulations of a ScramJet performed by Sandia National Laboratories in Livermore with two large scale complex computational models [29], and Climate-LS-21-34 and Climate-LS-21-50 that correspond to the maxima of daily precipitations, which have been predicted by Lawrence Livermore National Laboratory with a climate computational model [30]. Section 5.1 deals with the ScramJet database, Section 5.2 is devoted to the climate database, Section 5.3 deals with the convergence analysis with respect to the number n_{MC} of additional realizations, and finally, in Section 6, we present an analysis of the results that have been obtained.

The applications presented below are devoted to the identification of m^{opt} and ε^{opt} and not to other important analyses such as the convergence of the probability distribution of the additional



(a) (b)

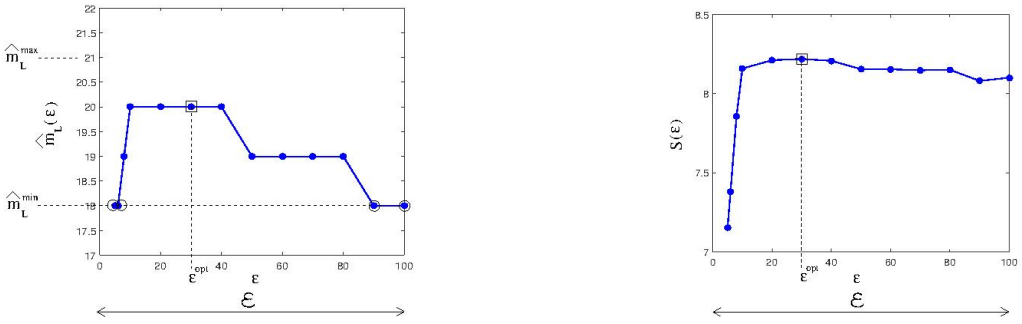
Figure 4: Helix database: For $\varepsilon = \varepsilon^{\text{opt}}$, graph of function $\alpha \mapsto \lambda_\alpha$ (left figure) and $\nu_{\text{sim}} = 40{,}000$ additional realizations of $\mathbf{X} = (X_1, X_2, X_3)$.

realizations constructed with probabilistic learning. Such analysis has been made in details in [1, 2].

5.1. Scramjet database

The database corresponds to two different numerical simulations referenced as Scramjet-d8 and Scramjet-d16. For these two cases, random vector $\mathbf{X}$ has dimension $n = 18$, which is constituted of $m_w = 11$ random parameters, $w_1, \ldots, w_{11}$, of the computational model and of $n_q = 7$ quantities of interest (QoI), $q = (q_1, \ldots, q_7)$, related to outputs of the computational model.

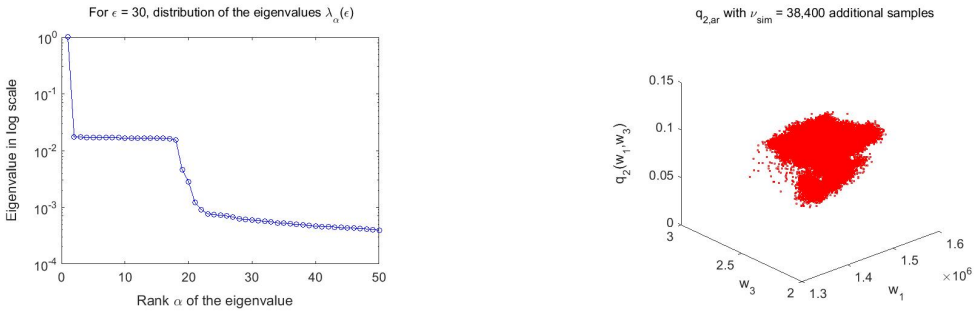
5.1.1. Scramjet-d8



(a) (b)

Figure 5: Scramjet-d8 database: graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ (left figure) and $\varepsilon \mapsto S(\varepsilon)$ (right figure), defined on $\mathcal{A}$. On the two figures, (i) $\mathcal{E} = \mathcal{A}$ is the set of the values of ε such that $\widehat{m}_L^{\min} \leq \widehat{m}_L(\varepsilon) \leq \widehat{m}_L^{\max}$, (ii) the values of $\widehat{m}_L(\varepsilon)$ surrounded by circle symbols correspond to the values of ε that minimize $m_L(\varepsilon)$ and that define the set $\mathcal{E}_0 \subset \mathcal{E} = \mathcal{A}$, and (iii) the square symbol localizes the optimal value ε^{opt} of ε.

For Scramjet-d8, there are $N = 256$ independent realizations in the database. The principal component analysis (step 2) yields $\nu = 17$ for which $\text{error}_{pca}(\nu) = 6.37 \times 10^{-8}$. The additional



(a) (b)

Figure 6: Scramjet-d8 database: For $\varepsilon = \varepsilon^{\text{opt}}$, graph of function $\alpha \mapsto \lambda_\alpha$ (left figure) and $\nu_{\text{sim}} = 38{,}400$ additional realizations of a QoI, q_2, as function of parameters w_1 and w_3.

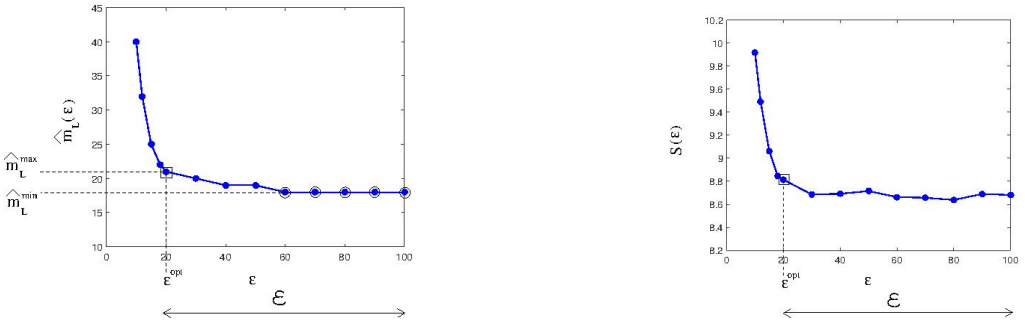
realizations are computed (step 7) with $n_{\text{MC}} = 150$ yielding $\nu_{\text{sim}} = 38{,}400$. The admissible finite set $\mathcal{A}$ of the values of ε is chosen to be $\{5, 6, 8, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$. The graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ and $\varepsilon \mapsto S(\varepsilon)$ defined on $\mathcal{A}$ are displayed in Fig. 5 (left and right). We have $\widehat{m}_L^{\min} = 18$, $\widehat{m}_L^{\max} = 21$, and $\mathcal{E}_0 = \{5, 6, 90, 100\} \subset \mathcal{E} = \mathcal{A}$. The entropy-based selection yields $\varepsilon^{\text{opt}} = 30$ and consequently, $\widehat{m}^{\text{opt}} = 20$, $m^{\text{opt}} = 19$. The relative mean-square error is $\text{error}_{proj}(\widehat{m}^{\text{opt}}) = 9.13 \times 10^{-3}$. For $\varepsilon = \varepsilon^{\text{opt}}$, Fig. 6(a) shows function $\alpha \mapsto \lambda_\alpha$ for which $\lambda_2 = 1.73 \times 10^{-2}$, $\lambda_{20} = 2.83 \times 10^{-3}$, and $\lambda_{21} = 1.22 \times 10^{-3}$. As an example of additional realizations generated by the probabilistic learning algorithm, Fig. 6(b) displays the 38,400 additional realizations of the QoI, q_2, as function of two parameters w_1 and w_3 that have been chosen for this illustration.

5.1.2. Scramjet-d16

For Scramjet-d16, there are $N = 172$ independent realizations in the database, $\nu = 17$, and $\text{error}_{pca}(\nu) = 1.57 \times 10^{-7}$. The additional realizations are computed with $n_{\text{MC}} = 150$ yielding $\nu_{\text{sim}} = 25{,}800$. Figures 7 (left and right) display the graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ and $\varepsilon \mapsto S(\varepsilon)$ defined on $\mathcal{A} = \{10, 12, 15, 18, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$. We have $\widehat{m}_L^{\min} = 18$, $\widehat{m}_L^{\max} = 21$, $\mathcal{E}_0 = \{60, 70, 80, 90, 100\} \subset \mathcal{E} = \{20, 30, 40, 50, 60, 70, 80, 90, 100\} \subset \mathcal{A}$. The entropy-based selection yields $\varepsilon^{\text{opt}} = 20$ and consequently, $\widehat{m}^{\text{opt}} = 21$, $m^{\text{opt}} = 20$, and $\text{error}_{proj}(\widehat{m}^{\text{opt}}) = 1.67 \times 10^{-2}$. For $\varepsilon = \varepsilon^{\text{opt}}$, Fig. 8(a) shows function $\alpha \mapsto \lambda_\alpha$ for which $\lambda_2 = 2.62 \times 10^{-2}$, $\lambda_{21} = 3.53 \times 10^{-3}$, and $\lambda_{22} = 2.48 \times 10^{-3}$. Similarly to the ScramJet-d8, Fig. 8(b) displays the 25,800 additional realizations of the QoI, q_2, as function of w_1 and w_3.

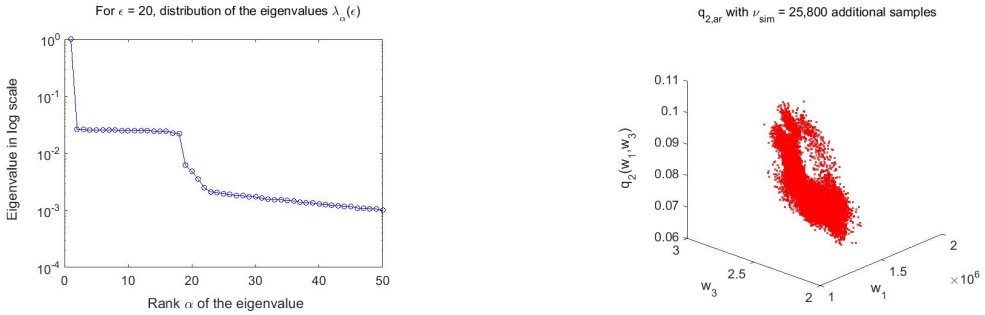
5.2. Climate database

The climate database is made up of $N = 130$ realizations of the precipitation magnitudes for each hour (24 hours per day) of 149 consecutive days, which have been performed with a climate computational model. The specific model used is the E3SM, formerly known as Accelerated Climate Modeling for Energy (ACME) [30]. Consequently, 130 realizations are available for $3,576$ consecutive hours. Two sets of data are extracted from this climate database, denoted by Climate-LS-21-34 and Climate-LS-21-50,which correspond to the maximum of daily precipitations for 14 consecutive days (from 21st day to 34th day) and for 30 consecutive days (from 21st day to 50th day), respectively.



(a) (b)

Figure 7: Scramjet-d16 database: graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ (left figure) and $\varepsilon \mapsto S(\varepsilon)$ (right figure), defined on $\mathcal{A}$. On the two figures, (i) $\mathcal{E} \subset \mathcal{A}$ is the subset of the values of ε such that $\widehat{m}_L^{\min} \leq \widehat{m}_L(\varepsilon) \leq \widehat{m}_L^{\max}$, (ii) the values of $\widehat{m}_L(\varepsilon)$ surrounded by circle symbols correspond to the values of ε that minimize $m_L(\varepsilon)$ and that define the set $\mathcal{E}_0 \subset \mathcal{E} \subset \mathcal{A}$, and (iii) the square symbol localizes the optimal value ε^{opt} of ε.

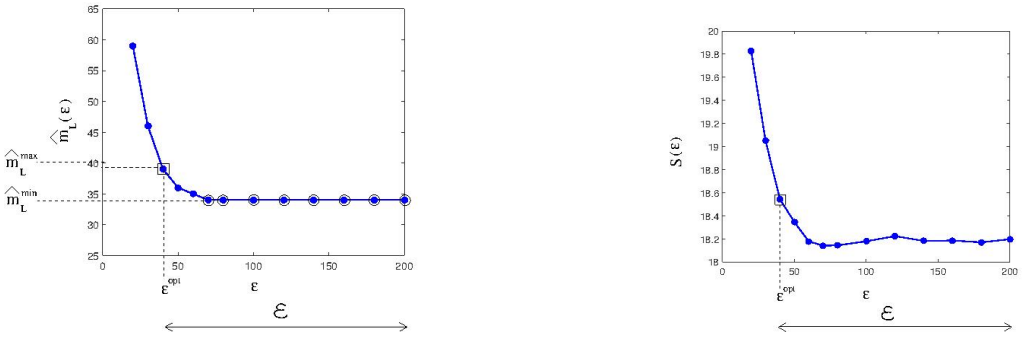


(a) (b)

Figure 8: Scramjet-d16 database: For $\varepsilon = \varepsilon^{\text{opt}}$, graph of function $\alpha \mapsto \lambda_\alpha$ (left figure) and $\nu_{\text{sim}} = 25{,}800$ additional realizations of a QoI, q_2, as function of parameters w_1 and w_3.

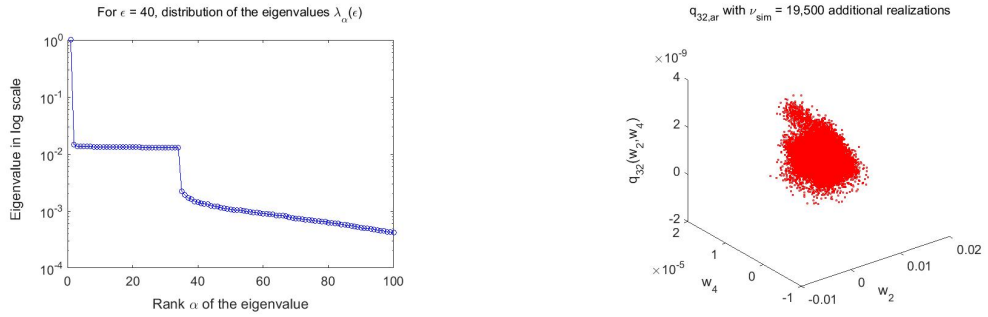
5.2.1. Climate-LS-21-34

For each one (indexed by ℓ) of the 14 consecutive days from the 21st day to the 34th day, the maximum p_ℓ of the precipitation is identified as well as the time t_ℓ at which this maximum occurs. Random vector $\mathbf{X}$ has dimension $n = 33$ and is constituted of $m_w = 5$ random parameters $w_1, \ldots, w_5$ of the computational model and $n_q = 28$ quantities of interest, $q = (q_1, \ldots, q_{28})$, made up of $t_1, \ldots, t_{14}$ and $p_1, \ldots, p_{14}$. The principal component analysis (step 2) yields $\nu = 33$ for which $\text{error}_{pca}(\nu) = 6.45 \times 10^{-16}$. The additional realizations are computed (step 7) with $n_{\text{MC}} = 150$ yielding $\nu_{\text{sim}} = 19{,}500$. The admissible finite set $\mathcal{A}$ of the values of ε is chosen to be $\{20, 30, 40, 50, 60, 70, 80, 100, 120, 140, 160, 180, 200\}$. The graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ and $\varepsilon \mapsto S(\varepsilon)$ defined on $\mathcal{A}$ are displayed in Fig. 9 (left and right). We have $\widehat{m}_L^{\min} = 34$, $\widehat{m}_L^{\max} = 40$, and $\mathcal{E}_0 = \{70, 80, 100, 120, 140, 160, 180, 200\} \subset \mathcal{E} = \{40, 50, 60, 70, 80, 100, 120, 140, 160, 180, 200\} \subset \mathcal{A}$. The entropy-based selection yields $\varepsilon^{\text{opt}} = 40$ and consequently, $\widehat{m}^{\text{opt}} = 39$, $m^{\text{opt}} = 38$, and $\text{error}_{proj}(\widehat{m}^{\text{opt}}) = 1.53 \times 10^{-2}$. For $\varepsilon = \varepsilon^{\text{opt}}$, Fig. 10(a) shows function $\alpha \mapsto \lambda_\alpha$ for which



(a) (b)

Figure 9: Climate-LS-21-34 database: graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ (left figure) and $\varepsilon \mapsto S(\varepsilon)$ (right figure), defined on $\mathcal{A}$. On the two figures, (i) $\mathcal{E} \subset \mathcal{A}$ is the subset of the values of ε such that $\widehat{m}_L^{\min} \leq \widehat{m}_L(\varepsilon) \leq \widehat{m}_L^{\max}$, (ii) the values of $\widehat{m}_L(\varepsilon)$ surrounded by circle symbols correspond to the values of ε that minimize $m_L(\varepsilon)$ and that define the set $\mathcal{E}_0 \subset \mathcal{E} \subset \mathcal{A}$, and (iii) the square symbol localizes the optimal value ε^{opt} of ε.



(a) (b)

Figure 10: Climate-LS-21-34: For $\varepsilon = \varepsilon^{\text{opt}}$, graph of function $\alpha \mapsto \lambda_\alpha$ (left figure) and $\nu_{\text{sim}} = 19{,}500$ additional realizations of a QoI q_{32} as function of parameters w_2 and w_4 (right figure).

$\lambda_2 = 1.46 \times 10^{-2}$, $\lambda_{39} = 1.47 \times 10^{-3}$, and $\lambda_{40} = 1.43 \times 10^{-3}$. Fig. 10(b) displays the cloud of the 19,500 additional realizations of QoI, q_{32}, as function of parameters w_2 and w_4.

5.2.2. Climate-LS-21-50

For each one (indexed by ℓ) of the 30 consecutive days from the 21st day to the 50th day, the maximum p_ℓ of the precipitation is identified as well as the time t_ℓ at which this maximum occurs. Random vector $\mathbf{X}$ has dimension $n = 65$ and is constituted of $m_w = 5$ random parameters $w_1, \ldots, w_5$ of the computational model and $n_q = 60$ quantities of interest, $q = (q_1, \ldots, q_{60})$, made up of $t_1, \ldots, t_{30}$ and $p_1, \ldots, p_{30}$, and $\nu = 65$ with $\text{error}_{pca}(\nu) = 7.12 \times 10^{-16}$. The additional realizations are computed with $n_{\text{MC}} = 150$ yielding $\nu_{\text{sim}} = 19{,}500$. Figures 11 (left and right) display the graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ and $\varepsilon \mapsto S(\varepsilon)$ defined on $\mathcal{A} = \{60, 80, 85, 90, 95, 100, 110, 120, 140, 160, 170, 180, 190, 200, 210, 220, 250\}$. We have $\widehat{m}_L^{\min} = 66$, $\widehat{m}_L^{\max} = 79$, and $\mathcal{E}_0 = \{140, 160, 170, 180, 190, 200, 210, 220, 250\} \subset \mathcal{E} =$

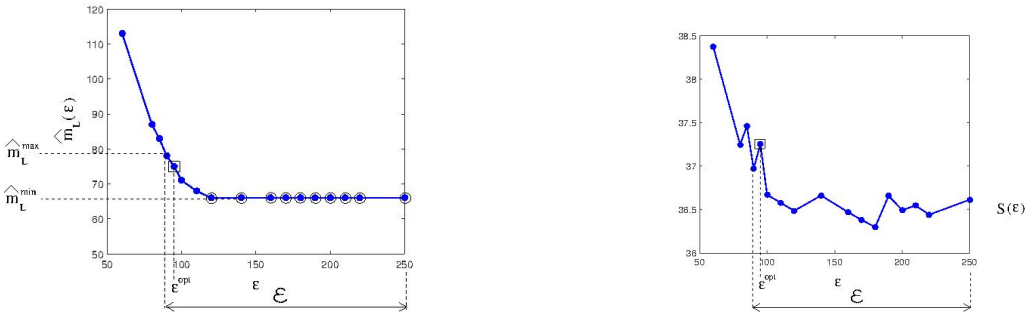


Figure 11: Climate-LS-21-50 database: graphs of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ (left figure) and $\varepsilon \mapsto S(\varepsilon)$ (right figure), defined on $\mathcal{A}$. On the two figures, (i) $\mathcal{E} \subset \mathcal{A}$ is the subset of the values of ε such that $\widehat{m}_L^{\min} \leq \widehat{m}_L(\varepsilon) \leq \widehat{m}_L^{\max}$, (ii) the values of $\widehat{m}_L(\varepsilon)$ surrounded by circle symbols correspond to the values of ε that minimize $m_L(\varepsilon)$ and that define the set $\mathcal{E}_0 \subset \mathcal{E} \subset \mathcal{A}$, and (iii) the square symbol localizes the optimal value ε^{opt} of ε.

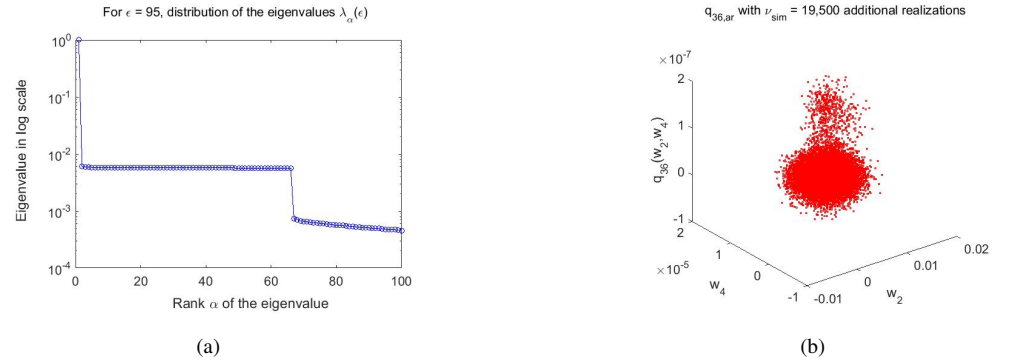
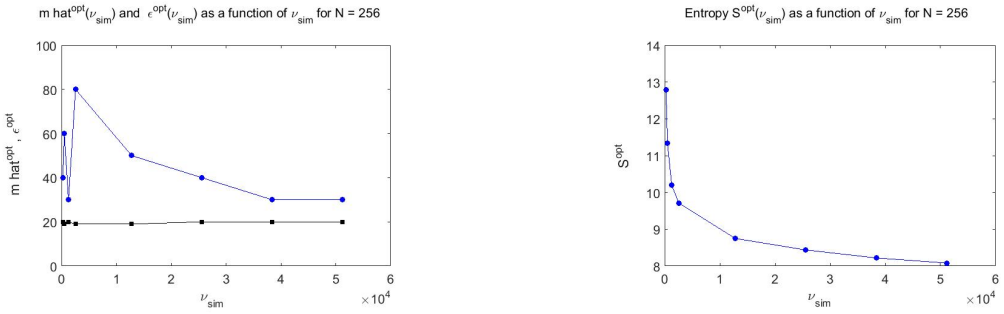


Figure 12: Climate-LS-21-50: For $\varepsilon = \varepsilon^{\text{opt}}$, graph of function $\alpha \mapsto \lambda_\alpha$ (left figure) and $\nu_{\text{sim}} = 19{,}500$ additional realizations of a QoI q_{36} as function of parameters w_2 and w_4 (right figure).

$\{90, 95, 100, 110, 120, 140, 160, 170, 180, 190, 200, 210, 220, 250\} \subset \mathcal{A}$. The entropy-based selection yields $\varepsilon^{\text{opt}} = 95$ and consequently, $\widehat{m}^{\text{opt}} = 75$, $m^{\text{opt}} = 74$. The relative mean-square error is $\text{error}_{proj}(\widehat{m}^{\text{opt}}) = 7.04 \times 10^{-4}$. For $\varepsilon = \varepsilon^{\text{opt}}$, Fig. 12(a) shows function $\alpha \mapsto \lambda_\alpha$ for which $\lambda_2 = 6.13 \times 10^{-3}$, $\lambda_{75} = 6.01 \times 10^{-4}$, and $\lambda_{76} = 1.43 \times 10^{-3}$. Fig. 12(b) displays the cloud of the 19,500 additional realizations of QoI, q_{36}, as function of parameters w_2 and w_4.

5.3. Convergence analysis with respect to the number $\nu_{\text{sim}} = N \times n_{\text{MC}}$ of additional realizations

The convergence of the optimal values is analyzed (see Section 3.2.3) studying the simple convergence in $\mathbb{R}$ of the sequences of real numbers $\{\varepsilon^{\text{opt}}(n_{\text{MC}})\}_{n_{\text{MC}}}$ and $\{\widehat{m}^{\text{opt}}(n_{\text{MC}})\}_{n_{\text{MC}}}$. Such a convergence analysis can also be presented as a function of the number $\nu_{\text{sim}} = N \times n_{\text{MC}}$ of additional realizations. We present the convergence analysis for the ScramJet-d8 database for which $N = 256$. The results are similar for the other three databases. Fig. 13(a) displays the graphs of functions $\nu_{\text{sim}} \mapsto \widehat{m}^{\text{opt}}(\nu_{\text{sim}})$ and $\nu_{\text{sim}} \mapsto \varepsilon^{\text{opt}}(\nu_{\text{sim}})$. Fig. 13(b) displays the graph of function $\nu_{\text{sim}} \mapsto S^{\text{opt}}(\nu_{\text{sim}})$ in which $S^{\text{opt}}(\nu_{\text{sim}}) = S(\varepsilon^{\text{opt}}(\nu_{\text{sim}}); \nu_{\text{sim}})$ (note that the entropy S depends not



(a) (b)

Figure 13: ScramJet-d8 database: convergence analysis of the optimal solution as a function of the number $\nu_{\text{sim}} = N \times n_{\text{MC}}$ of additional realizations. (a): graphs of functions $\nu_{\text{sim}} \mapsto \widehat{m}^{\text{opt}}(\nu_{\text{sim}})$ (black thin line) and $\nu_{\text{sim}} \mapsto \varepsilon^{\text{opt}}(\nu_{\text{sim}})$ (blue thick line). (b): graph of the entropy function $\nu_{\text{sim}} \mapsto S^{\text{opt}}(\nu_{\text{sim}})$ for the optimal value $\varepsilon^{\text{opt}}(\nu_{\text{sim}})$ of ε.

only on ε but also of the number of realizations, ν_{sim}). It can be seen that the convergence is obtained for $\nu_{\text{sim}} = 38{,}400$, that is to say for $n_{\text{MC}} = 150$.

Remark. Figs. 6(b) and 8(b) (graph of $(w_1, w_3) \mapsto q_2(w_1, w_3)$) and 10(b), and 12(b) (graph of $(w_2, w_4) \mapsto q_2(w_2, w_4)$) are merely given as illustration. These graphs suggest the existence of an intrinsic data structure and the ability to generate realizations according to this structure (the generated data clouds exhibit clustering).

6. Analysis of the results

Figs. 4, 6, 8, 10, and 12 (left figures), which display the graphs of $\alpha \mapsto \lambda_\alpha(\varepsilon)$ for the five databases, show that value 0.1 proposed for parameter L (in the criterion defined by Eq. (1)) is well chosen. Figs. 3, 5, 7, 9, and 11 show that the variations of functions $\varepsilon \mapsto \widehat{m}_L(\varepsilon)$ (left figures) and $\varepsilon \mapsto S(\varepsilon)$ (right figures) are relatively different from a database to another one. In particular, the entropy function $\varepsilon \mapsto S(\varepsilon)$ is not monotonic and Fig. 5(b) is not similar, for instance, to Fig. 11(b). Figures 3, 5, 7, 9, and 11 (right figures) also show that the value 0.2 that is proposed for parameter χ in Eq. (6) is well adapted in order to obtain a sufficiently robust algorithm for the five databases that have been analyzed). Following the second remark introduced in Section 3.1, these figures show that the entropy $S(\varepsilon)$ effectively "grows on average" with $m = m_L(\varepsilon)$. Finally, as examples, Figs. 4, 6,8, 10, and 9 (right figures) display the clouds of the additional realizations generated by the probabilistic learning algorithm. These figures show the complexity of the discovered geometry of the subset around which the probability measure is concentrated. In addition, the additional realizations stay concentrated around the geometry of the clouds made up of the initial data points. Using the optimal values ε^{opt} and m^{opt} of ε and m, the MCMC generator proposed does not induce scattering.

7. Conclusions

In this paper, we have presented a selection model of the isotropic-diffusion kernel hyperparameter of the diffusion maps that is used in a probabilistic learning model on manifold recently

proposed by the authors. The selection model has been tested on several databases and seems to be independent of the databases used. The probabilistic learning on manifold that has been proposed is algebraically closed with respect to its hyperparameters and which can be applied for some databases. However, this robustness that has been found must be still confirmed using another databases.

8. Acknowledgments

Part of this research was supported by the ScramJet-UQ project funded under DARPA's EQUIPS Program. The authors thank the Combustion Research Facility of Sandia National Laboratory in Livermore for the use of their simulated data of the ScramJet and the Climate Modeling and Analysis Group of Lawrence Livermore National Laboratory for the use of their simulated climate data in order to test the proposed methodology. Part of this research was also supported as part of the Energy Exascale Earth System Model (E3SM) project, funded by the U.S. Department of Energy (DOE), Office of Science, Office of Biological and Environmental Research under the auspices of the U.S. DOE by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. This research used resources of the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. DOE under contract DE-AC02-06CH11357.

Appendix A. Appendix A

In this appendix, we summarize the algorithm used in Step 7 for generating n_{MC} additional realizations $[z_{\text{ar}}^1], \ldots, [z_{\text{ar}}^{n_{\text{MC}}}]$ of random matrix $[\boldsymbol{Z}^{\varepsilon,m}]$, using the reduced ISDE for which the details can be found in [1]. Let $M = n_{\text{MC}} \times M_0$ be the positive integer in which M_0 is a positive integer greater or equal to 1. The reduced-order ISDE is solved on the finite interval $\mathcal{R} = [0\,, M\,\Delta r]$, in which Δr is the sampling step of the continuous index parameter r. The integration scheme is the Störmer-Verlet scheme for which $M+1$ sampling points $\{r_\ell, \ell = 0, \ldots, M\}$ are used with $r_\ell = \ell\,\Delta r$. For $\ell = 0, \ldots, M-1$, the algorithm is written as

$$[\boldsymbol{\mathcal{Z}}_{\ell+\frac{1}{2}}^{\varepsilon,m}] = [\boldsymbol{\mathcal{Z}}_{\ell}^{\varepsilon,m}] + \frac{\Delta r}{2}\,[\boldsymbol{\mathcal{Y}}_{\ell}^{\varepsilon,m}]\,, \tag{A.1}$$

$$[\boldsymbol{\mathcal{Y}}_{\ell+1}^{\varepsilon,m}] = \frac{1-\beta}{1+\beta}\,[\boldsymbol{\mathcal{Y}}_{\ell}^{\varepsilon,m}] + \frac{\Delta r}{1+\beta}\,[\boldsymbol{\mathcal{L}}_{\ell+\frac{1}{2}}] + \frac{\sqrt{f_0}}{1+\beta}\,[\Delta\boldsymbol{\mathcal{W}}_{\ell+1}]\,, \tag{A.2}$$

$$[\boldsymbol{\mathcal{Z}}_{\ell+1}^{\varepsilon,m}] = [\boldsymbol{\mathcal{Z}}_{\ell+\frac{1}{2}}^{\varepsilon,m}] + \frac{\Delta r}{2}\,[\boldsymbol{\mathcal{Y}}_{\ell+1}^{\varepsilon,m}]\,. \tag{A.3}$$

The quantities in these equations are defined as follows.

(i) For $\ell = 0$, a given realization of the random matrix $[\boldsymbol{\mathcal{Z}}_0^{\varepsilon,m}]$ is known and is denoted by $[\eta_d]$. The matrix $[\boldsymbol{\mathcal{Y}}_0^{\varepsilon,m}]$ is written as $[\boldsymbol{\mathcal{Y}}_0^{\varepsilon,m}] = [\boldsymbol{\mathcal{N}}]\,[a]$ in which $[a] = [g]\,([g]^T\,[g])^{-1} \in \mathbb{M}_{N,m}$ and where $[\boldsymbol{\mathcal{N}}]$ is a random matrix with values in $\mathbb{M}_{\nu,N}$ for which its columns are N independent copies of a normalized Gaussian vector with values in $\mathbb{R}^\nu$.

(ii) For $\ell = 0, \ldots, M-1$, we have $[\Delta\boldsymbol{\mathcal{W}}_{\ell+1}] = [\Delta\mathbb{W}_{\ell+1}]\,[a]$, in which $[\Delta\mathbb{W}_1], \ldots, [\Delta\mathbb{W}_M]$ are M independent random matrices with values in $\mathbb{M}_{\nu,N}$ and where, for all $k = 1, \ldots, \nu$ and $j = 1, \ldots, N$, the real-valued random variables $\{[\Delta\mathbb{W}_{\ell+1}]_{kj}\}_{kj}$ are independent, Gaussian, second-order, and centered such that $E\{[\Delta\mathbb{W}_{\ell+1}]_{kj}[\Delta\mathbb{W}_{\ell+1}]_{k'j'}\} = \Delta r\,\delta_{kk'}\,\delta_{jj'}$.

(iii) We have $\beta = f_0\,\Delta r\,/4$ and $[\boldsymbol{\mathcal{L}}_{\ell+\frac{1}{2}}]$ is the $\mathbb{M}_{\nu,m}$-valued random variable such that

$$[\boldsymbol{\mathcal{L}}_{\ell+\frac{1}{2}}] = [\mathcal{L}([\mathcal{Z}^{\varepsilon,m}_{\ell+\frac{1}{2}}])] = [L([\mathcal{Z}^{\varepsilon,m}_{\ell+\frac{1}{2}}]\,[g]^T)]\,[a]\,. \tag{A.4}$$

For all $[u] = [\mathbf{u}^1 \ldots \mathbf{u}^N]$ in $\mathbb{M}_{\nu,N}$ with $\mathbf{u}^\ell = (u^\ell_1, \ldots, u^\ell_\nu)$ in $\mathbb{R}^\nu$, the matrix $[L([u])]$ in $\mathbb{M}_{\nu,N}$ is defined, for all $k = 1, \ldots, \nu$ and for all $\ell = 1, \ldots, N$, by

$$[L([u])]_{k\ell} = \frac{1}{p(\mathbf{u}^\ell)}\,\{\boldsymbol{\nabla}_{\mathbf{u}^\ell}\,p(\mathbf{u}^\ell)\}_k\,, \tag{A.5}$$

$$p(\mathbf{u}^\ell) = \frac{1}{N}\sum_{j=1}^{N} \exp\{-\frac{1}{2\widehat{s}_\nu^{\,2}}\|\frac{\widehat{s}_\nu}{s_\nu}\boldsymbol{\eta}^j - \mathbf{u}^\ell\|^2\}\,, \tag{A.6}$$

$$\boldsymbol{\nabla}_{\mathbf{u}^\ell}\,p(\mathbf{u}^\ell) = \frac{1}{\widehat{s}_\nu^{\,2}}\frac{1}{N}\sum_{j=1}^{N}(\frac{\widehat{s}_\nu}{s_\nu}\boldsymbol{\eta}^j - \mathbf{u}^\ell)\,\exp\{-\frac{1}{2\widehat{s}_\nu^{\,2}}\|\frac{\widehat{s}_\nu}{s_\nu}\boldsymbol{\eta}^j - \mathbf{u}^\ell\|^2\}\,, \tag{A.7}$$

$$s_\nu = \left\{\frac{4}{N(2+\nu)}\right\}^{1/(\nu+4)} \quad , \quad \widehat{s}_\nu = \frac{s_\nu}{\sqrt{s_\nu^2 + \frac{N-1}{N}}}\,. \tag{A.8}$$

Introducing $\rho = M_0\,\Delta r$, the n_{MC} additional realizations $[z^1_{\mathrm{ar}}], \ldots, [z^{n_{\mathrm{MC}}}_{\mathrm{ar}}]$ of random matrix $[\mathbf{Z}^{\varepsilon,m}]$ are given by

$$[z^\ell_{\mathrm{ar}}] = [\mathcal{Z}^{\varepsilon,m}_{\ell\times\rho}(\theta)] \quad , \quad \ell = 1, \ldots, n_{\mathrm{MC}}\,, \tag{A.9}$$

in which $[\mathcal{Z}^{\varepsilon,m}_{\ell\times\rho}(\theta)]$ denotes the deterministic solution of any realization θ of Eqs. (A.1) to (A.3).

- If $M_0 = 1$, then $\rho = \Delta r$ and the n_{MC} additional realizations are dependent, but the ergodic property of $\{[\mathcal{Z}^{\varepsilon,m}_\ell]\}_\ell$ can be invoked for ensuring the convergence of statistics constructed using $[z^1_{\mathrm{ar}}], \ldots, [z^{n_{\mathrm{MC}}}_{\mathrm{ar}}]$ for random matrix $[\mathbf{Z}^{\varepsilon,m}]$.

- If integer M_0 is chosen sufficiently large (such that ρ is much larger than the relaxation time of the dissipative Hamiltonian dynamical system), then $[z^1_{\mathrm{ar}}], \ldots, [z^{n_{\mathrm{MC}}}_{\mathrm{ar}}]$ can approximately be considered as independent realizations of random matrix $[\mathbf{Z}^{\varepsilon,m}]$.

[1] C. Soize, R. Ghanem, Data-driven probability concentration and sampling on manifold, Journal of Computational Physics 321 (2016) 242–258. doi:10.1016/j.jcp.2016.05.044.

[2] C. Soize, R. Ghanem, Polynomial chaos representation of databases on manifolds, Journal of Computational Physics 335 (2017) 201–221. doi:10.1016/j.jcp.2017.01.031.

[3] B. Schölkopf, A. Smola, K. Müller, Kernel principal component analysis, in: W. Gerstner, A. Germond, M. Hasler, J. Nicoud (Eds.), Artificial Neural Networks ICANN'97, Vol. 1327 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 583–588.

[4] V. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 2000.

[5] C. C. Aggarwal, C. Zhai, Mining Text Data, Springer Science & Business Media, 2012.

[6] A. S. Dalalyan, A. B. Tsybakov, Sparse regression learning by aggregation and langevin monte-carlo, Journal of Computer and System Sciences 78 (5) (2012) 1423–1443.

[7] K. P. Murphy, Machine Learning: A Probabilistic Perspective, MIT press, 2012.

[8] M.-F. F. Balcan, V. Feldman, Statistical active learning algorithms, in: Advances in Neural Information Processing Systems, 2013, pp. 1295–1303.

[9] G. James, D. Witten, T. Hastie, R. Tibshirani, An Introduction to Statistical Learning, Vol. 112, Springer, 2013.

[10] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, W. Zhang, Knowledge vault: A web-scale approach to probabilistic knowledge fusion, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2014, pp. 601–610.

[11] Z. Ghahramani, Probabilistic machine learning and artificial intelligence, Nature 521 (7553) (2015) 452.
[12] J. Taylor, R. J. Tibshirani, Statistical learning and selective inference, Proceedings of the National Academy of Sciences 112 (25) (2015) 7629–7634.
[13] R. Ghanem, D. Higdon, H. Owhadi, Handbook of Uncertainty Quantification, Vol. 1 to 3, Springer, Cham, Switzerland, 2017. doi:10.1007/978-3-319-12385-1.
[14] D. Jones, M. Schonlau, W. Welch, Efficient global optimization of expensive black-box functions, Journal of Global Optimization 13 (4) (1998) 455–492.
[15] N. Queipo, R. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, K. Tucker, Surrogate-based analysis and optimization, Progress in Aerospace Science 41 (1) (2005) 1–28. doi:10.1016/j.paerosci.2005.02.001.
[16] R. Byrd, G. Chin, W. Neveitt, J. Nocedal, On the use of stochastic Hessian information in optimization methods for machine learning, SIAM Journal of Optimization 21 (3) (2011) 977–995. doi:10.1137/10079923X.
[17] T. Homem-de Mello, G. Bayraksan, Monte Carlo sampling-based methods for stochastic optimization, Surveys in Operations Researh and Management Science 19 (1) (2014) 56–85. doi:10.1016/j.sorms.2014.05.001.
[18] J. Kleijnen, W. van Beers, I. van Nieuwenhuyse, Constrained optimization in expensive simulation: Novel approach, European Journal of Operational Research 202 (1) (2010) 164–174. doi:10.1016/j.ejor.2009.05.002.
[19] Z. Wang, M. Zoghi, F. Hutter, D. Matheson, N. de Freitas, Bayesian optimization in a billion dimensions via random embeddings, Journal of Artificial Intelligence Research 55 (2016) 361–387. doi:10.1613/jair.4806.
[20] J. Xie, P. Frazier, S. Chick, Bayesian optimization via simulation with pairwise sampling and correlated pair beliefs, Operations Research 64 (2) (2016) 542–559.
[21] X. Du, W. Chen, Sequential optimization and reliability assessment method for efficient probabilistic design, ASME Journal of Mechanical Design 126 (2) (2004) 225–233. doi:10.1115/1.1649968.
[22] M. Eldred, Design under uncertainty employing stochastic expansion methods, International Journal for Uncertainty Quantification 1 (2) (2011) 119–146. doi:10.1615/Int.J.UncertaintyQuantification.v1.i2.20.
[23] W. Yao, X. Chen, W. Luo, M. vanTooren, J. Guo, Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles, Progress in Aerospace Sciences 47 (2011) 450–479.
[24] R. Ghanem, C. Soize, Probabilistic nonconvex constrained optimization with fixed number of function evaluations, International Journal for Numerical Methods in Engineering *Published on line 2017*. doi:10.1002/nme.5632.
[25] R. Ghanem, C. Soize, C.-R. Thimmisetty, Optimal well-placement using a probabilistic learning, Data-Enabled Discovery and Applications , Accepted for publication, December 20, 2017.
[26] C. Thimmisetty, P. Tsilifis, R. Ghanem, Homogeneous chaos basis adaptation for design optimization under uncertainty: Application to the oil well placement problem, Artificial Intelligence for Engineering Design, Analysis and Manufacturing 31 (3 (Uncertainty Quantification for Engineering Design)) (2017) 265–276. doi:10.1017/S0890060417000166.
[27] E. Roesler, M. Taylor, W. Lin, Q. Tang, S. Klein, Climatology of the accelerated climate model for energy's community atmospheric model, version 5, configured with variable resolution over the continental united states, Geosci. Model Dev. in preparation.
[28] C. Terai, P. Caldwell, S. Klein, Q. Tang, M. Branstetter, The atmospheric hydrologic cycle in the acme v0.3 model, Clim Dyn (2017) doi:10.1007/s00382–017–3803–x.
[29] C. Soize, R. Ghanem, C. Safta, X. Huan, Z. V. anf J. Oefelein, G. Lacaze, H. Najm, Enhancing model predictability for a scramjet using probabilistic learning on manifold, AIAA Journal 2018.
[30] D. Bader, W. Collins, R. Jacob, P. Jones, P. Rasch, M. Taylor, P. Thornton, D. Williams, D. Bader, W. Collins, R. Jacob, P. Jones, P. Rasch, M. Taylor, P. Thornton, D. Williams, Accelerated climate modeling for energy (acme) project strategy and initial implementation plan, Tech. rep. (2014).
URL `https://climatemodeling.science.energy.gov/sites/default/files/publications/acme-project-strategy-plan.pdf`
[31] C. Soize, Polynomial chaos expansion of a multimodal random vector, SIAM/ASA Journal on Uncertainty Quantification 3 (1) (2015) 34–60. doi:10.1137/140968495.
[32] A. Bowman, A. Azzalini, Applied Smoothing Techniques for Data Analysis, Oxford University Press, Oxford, UK, 1997.
[33] D. Scott, Multivariate Density Estimation: Theory, Practice, and Visualization, 2nd Edition, John Wiley and Sons, New York, 2015.
[34] C. Soize, Construction of probability distributions in high dimension using the maximum entropy principle. applications to stochastic processes, random fields and random matrices, International Journal for Numerical Methods in Engineering 76 (10) (2008) 1583–1611. doi:10.1002/nme.2385.
[35] M. Girolami, B. Calderhead, Riemann manifold Langevin and Hamiltonian Monte Carlo methods, Journal of the Royal Statistics Society 73 (2) (2011) 123–214. doi:10.1111/j.1467-9868.2010.00765.x.
[36] R. Neal, MCMC using hamiltonian dynamics, in: S. Brooks, A. Gelman, G. Jones, X.-L. Meng (Eds.), Handbook of Markov Chain Monte Carlo, Chapman and Hall-CRC Press, Boca Raton, 2011, Ch. 5. doi:10.1201/b10905-6.
[37] J. Spall, Introduction to Stochastic Search and Optimization, Wiley-Interscience, 2003.

[38] R. Coifman, S. Lafon, A. Lee, M. Maggioni, B. Nadler, F. Warner, S. Zucker, Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps, PNAS 102 (21) (2005) 7426–7431. doi:10.1073/pnas.0500334102.
[39] R. Coifman, S. Lafon, Diffusion maps, applied and computational harmonic analysis, Applied and Computational Harmonic Analysis 21 (1) (2006) 5–30. doi:10.1016/j.acha.2006.04.006.
[40] E. T. Jaynes, Information theory and statistical mechanics, Physical Review 106 (4) (1957) 620–630.
[41] E. T. Jaynes, Information theory and statistical mechanics, Physical Review 108 (2) (1957) 171–190.
[42] R. M. Gray, Entropy and Information Theory, 2nd Edition, Springer, New York, 2011. doi:10.1007/978-1-4419-7970-4.
[43] C. Soize, Uncertainty Quantification. An Accelerated Course with Advanced Applications in Computational Engineering, Springer, New York, 2017. doi:10.1007/978-3-319-54339-0.
[44] C. E. Shannon, A mathematical theory of communication, The Bell System Technical Journal 27 (1948) 379–423 and 623–659.