Identification of input random field samples causing extreme responses

Wayne Isaac T. Uy[1*], Mircea D. Grigoriu[1,2]
[1]Center for Applied Mathematics
[2]Department of Civil and Environmental Engineering
{wtu4,mdg12}@cornell.edu
Cornell University

Abstract

Consider a physical system modeled by a differential equation that depends on a coefficient random field. The objective of this work is to identify samples of this random field which yield extreme response as a means to: study the law of the input conditioned on rare events and predict if a random field sample causes such an event. This differs from reliability engineering which focuses on computation of failure probabilities. We investigate two classification schemes that identify these samples of interest: physics-based indicators which are functionals of the input random field and surrogate models which approximate the response. As an alternative to these approaches, we propose a general framework consisting of two stages that combines the use of a physics-based surrogate model and a machine learning classifier. In the first stage, a multifidelity surrogate that requires infrequent evaluations of the full model is designed. This surrogate is then used to generate a sufficient number of samples of random fields that yield extreme events to train a machine learning classifier in the second stage. We study the analytical properties required of the surrogate model and demonstrate through numerical examples the synergy of the proposed approach.

Keywords— rare event sampling, uncertainty quantification, surrogate models, extreme events, support vector machines

1 Introduction

We motivate this work with a simple physical example involving a unit length elastic rod in 1-dimension where one end is pulled by a time-dependent force. Suppose that the elasticity coefficient can be modeled by a random field $A(x,\omega)$ defined on the probability space $(\Omega,\mathcal{F},P)$ where $\omega\in\Omega$ and $x\in[0,1]$ refers to a location on the rod. For $t>0$, denote by $U(t,x,\omega)\in\mathbb{R}$ the displacement field whose dynamics are described by the Euler-Lagrange equation

$$\frac{\partial^2 U(t,x,\omega)}{\partial t^2}=\frac{\partial}{\partial x}\left(A(x,\omega)\frac{\partial}{\partial x}U(t,x,\omega)\right),\qquad x\in[0,1],\ t>0,\ \omega\in\Omega \tag{1.1}$$

*corresponding author

subject to initial and boundary conditions:

$$U(t,0,\omega)=0, \qquad A(1,\omega)\left(\int_0^1 \frac{dy}{A(y,\omega)}\right)^{-1}\frac{\partial}{\partial x}U(t,1,\omega)=at, \qquad t>0,\, a>0 \tag{1.2}$$

$$U(0,x,\omega)=0, \qquad \frac{\partial}{\partial t}U(0,x,\omega)=\left(a\int_0^x \frac{dy}{A(y,\omega)}\right)\left(\int_0^1 \frac{dy}{A(y,\omega)}\right)^{-1}, \qquad x\in[0,1].$$

Oftentimes, we are interested in quantities of interest (QoI) $Q(\omega)\in\mathbb{R}$ which are functionals of the random field $U(t,x,\omega)$.

Literature in reliability engineering and rare event simulation is concerned with the computation of the (failure) probability $P(Q>\tau)$ where for large τ, $\{Q>\tau\}$ is a rare event. Commonly employed methods [1] to compute such quantities express Q as a function of the random vector $Z(\omega)\in\mathbb{R}^d$, i.e. $Q(Z(\omega))$, and include the following. Asymptotic approaches like the first-order reliability method locate the point z^* on the limit state $\{z\in\mathbb{R}^d \,|\, Q(z)=\tau\}$ with the largest likelihood based on approximations of first and second order Taylor expansion of the limit state centered at z^*. Importance sampling makes use of a change in measure so that the failure probability is approximated by weighted samples in the failure domain $F=\{z\in\mathbb{R}^d \,|\, Q(z)>\tau\}$. Subset simulation iteratively considers a decreasing sequence of sets, each containing F. Finally, surrogate models replace Q by an inexpensive approximation $\widetilde{Q}$ afterwhich Monte Carlo simulation is invoked to compute probabilities. Examples of surrogate models used for this purpose include stochastic Galerkin [2], sparse grid stochastic collocation [3], and support vector machines (SVM) [4].

While failure probabilities are useful in assessing risks associated with structures, materials, etc., they provide limited if any information for controlling and mitigating these risks. They do not offer insight on how to design these objects better to avoid undesired events of low probability. In this work, we therefore deviate from the existing objectives of reliability engineering and rare event simulation. Instead, we focus on identifying samples $A(x,\omega)$ of A which cause large samples of Q, i.e. $Q(\omega)>\tau$ for large τ, by examining various classification schemes below which determine whether or not a sample $A(x,\omega)$ yields extreme events. Through a classifier, a sufficient number of samples of $A(x,\omega)\,|\,Q(\omega)>\tau$ can be obtained that enables one to: study the conditional statistics and distribution of $A(x,\omega)\,|\,Q(\omega)>\tau$, understand why rare events occur in such systems, and predict the occurrence of (unobserved) rare events. The relevance of this problem extends beyond the field of reliability engineering and includes applications such as detecting rare defects in additive manufacturing (agile production) of materials [5]. The shift in objective implies that the tools we require are different from those developed in reliability engineering.

We motivate our problem by first examining existing work. The only studies we are aware of that are pertinent to our objectives are [6, 7]. An extensive survey of how system dynamics causes rare bursts in the values of the state variables is detailed in [7]. In [6], the authors investigated a data-driven approach to identify what induces bursts in the energy dissipation rate of the Kolmogorov flow, modeled by an inrompressible Navier-Stokes equation exhibiting chaotic behavior. An indicator that is a functional of the velocity field was deduced from solving an optimization problem that relies on the defining equations of the system, i.e. it is physics-based. It was later validated using a large number of data based on statistics such as the rate of successful predictions and rejections. To investigate the feasibility of this approach, we also explore a classifier through the use of physics-based indicators – quantities computable from $A(x,\omega)$ which signal $Q(\omega)>\tau$. However, we demonstrate that even in the simplest of systems, indicators may be challenging to deduce analytically and are computationally expensive to calibrate due to the infrequent occurrence of rare events. The analysis undertaken only emphasizes the difficulty of the problem and the limitations of existing methods.

We therefore seek a classifier that is universal instead of an indicator that is problem-dependent. The computational cost of the indicator mentioned above motivates the use of a surrogate model $\widetilde{Q}$ that approximates Q; it then acts as a classifier by checking if $\widetilde{Q}>\tau$ or otherwise. Although a vast array of efficient surrogate models [8] have been proposed for reliability engineering that only require a small number of model evaluations, they are insufficient for our problem since they address a different objective. Specifically,

reliability engineering methods are geared to obtain convergence in failure probability estimates so that $P(Q > \tau) \approx P(\widetilde{Q} > \tau)$. This does not guarantee that the measurable sets $\{\omega \,|\, Q(\omega) > \tau\}$ and $\{\omega \,|\, \widetilde{Q}(\omega) > \tau\}$ are close approximations of each other which we later confirm with an example. Rather, our objective above mathematically translates to ensuring that the discrepancy between these two sets is minimized. We also note that using surrogate models alone still presents challenges in rare event simulation. Machine learning classifiers, for instance, require a sufficient number of samples in the low probability regions (i.e. failure domain) for training. Also, [2] underscores that surrogate models may over or underestimate probabilities of failure.

In line with our objectives, we build on the above concerns by proposing a two-stage approach which is computationally efficient and probabilistically accurate and that leverages on the strengths of different types of classifiers. In the first stage, we formulate a multifidelity physics-based surrogate model in the spirit of [2, 3]. The surrogate model is physics-based in the sense that it is constructed using the defining equations of the physical system. The multifidelity surrogate comprises the computationally feasible surrogate model $\widetilde{Q}$ and infrequent solves of the expensive full model Q which guarantee that $Q > \tau$ whenever $\widetilde{Q} > \tau$. In the second stage, samples of $A(x, \omega)$ are then generated using the multifidelity surrogate which ensures that a large number of samples of the rare event $A(x, \omega) \,|\, Q(\omega) > \tau$ is present to accurately train a machine learning classifier. Unlike the surrogate, this classifier only considers the data and does not incorporate the system equations anymore. The classifier then serves as a mechanism to predict whether $Q > \tau$ from data. Compared to physics-based indicators, the classification scheme in the proposed framwework is not problem-dependent.

The sections below detail the various classification approaches we investigate. Section 2 motivates the proposed approach by attempting to discover physics-based indicators that signal rare events using analytical arguments and later supported by data. Due to the challenges of validating and calibrating indicators, Section 3 proposes a multifidelity surrogate approximation for the QoI as an alternative classification scheme. Finally, the two-stage approach that combines the multifidelity surrogate and machine learning classifiers is tackled in Section 4.

2 Physics-based indicators for rare events

In this section, we present our problem formulation and motivate the framework we propose in Section 4 by first investigating existing alternatives for classification and prediction of rare events. We attempt to deduce indicators from $A(x, \omega)$ which signal $Q(\omega) > \tau$ for large τ in the spirit of [6, 7]. The case when the random field $A(x, \omega)$ is represented by an infinite-dimensional noise model and a finite-dimensional noise model are addressed in Section 2.1 and 2.2, respectively. The following discussion outlines the difficulty of obtaining indicators for simple systems and that a large number of samples of the rare event are needed to validate them. These limitations suggest why a different approach has to be sought.

2.1 Infinite-dimensional noise model

The physical example of the elastic rod in (1.1), (1.2) is revisited where the elasticity coefficient $A(x, \omega)$ is modeled by a translation Gaussian process.

Example 1 *Suppose that* $A(x, \omega) = \alpha + (\beta - \alpha) \cdot F^{-1}(\Phi(G(x, \omega)))$, $x \in [0, 1]$, $\beta \geq \alpha > 0$ *where* F *is the cumulative distribution function (cdf) of a* $Beta(p, q)$ *random variable,* Φ *is the cdf of the standard normal distribution, and* $G(x, \omega)$ *is a homogeneous zero-mean unit-variance Gaussian process with* $E[G(x, \omega) \cdot G(y, \omega)] = e^{-\lambda|x-y|}$, $x, y \in [0, 1]$. *It can be shown that the analytical solution to* (1.1), (1.2) *is given by*

$$U(t, x, \omega) = \left(at \int_0^x \frac{dy}{A(y, \omega)} \right) \left(\int_0^1 \frac{dy}{A(y, \omega)} \right)^{-1}, \ t > 0, \ x \in [0, 1]. \tag{2.1}$$

The quantity of interest we study is the random variable

$$Q_{max}(\omega) = \max_{x\in[0,1]} |U(1,x,\omega) - u(1,x)| \tag{2.2}$$

where $u(t,x) = atx$ is the deterministic continuum mechanics solution obtained from (2.1) if $A(x,\omega)$ were constant. We seek indicators from samples of $A(x,\omega)$ which signal $Q_{max} > \tau$ for large τ.

The simulations carried out below are based on $\alpha = 1, \beta = 10, p = 1, q = 3, \lambda = 2, a = 1$, and $\tau = 0.32$ with a spatial discretization of $\Delta x = 0.005$ to evaluate (2.1). The value of τ is chosen so that $P(Q_{max} > \tau) \approx 0.0014$. Note that A is a homogeneous random field with Beta marginal distribution of support (α, β) and correlation function similar to that of G. The parameter values are chosen just for the purposes of simulation and the analysis and conclusions below are independent of these values. Shown in the left panel of Figure 1 are 10000 samples of Q_{max} while the middle and the right panels display 5 samples of $A(x,\omega)$ and 5 samples of $A(x,\omega)\,|\,Q_{max}(\omega) > \tau$, respectively. Observe that samples of $A(x,\omega)\,|\,Q_{max}(\omega) > \tau$ manifest a common trend: they are generally increasing or decreasing. This behavior is consistent with plots of Figure 2 which present histograms of $A(x,\omega)|Q_{max}(\omega) > \tau$ using 10000 samples at various spatial locations $x \in [0,1]$ with the range of $A(x,\omega)$ rescaled from $[\alpha,\beta]$ to $[0,1]$. For comparison, the probability density function (pdf) of $A(x,\omega)$ for all $x \in [0,1]$ which is $Beta(p,q)$ is also included.

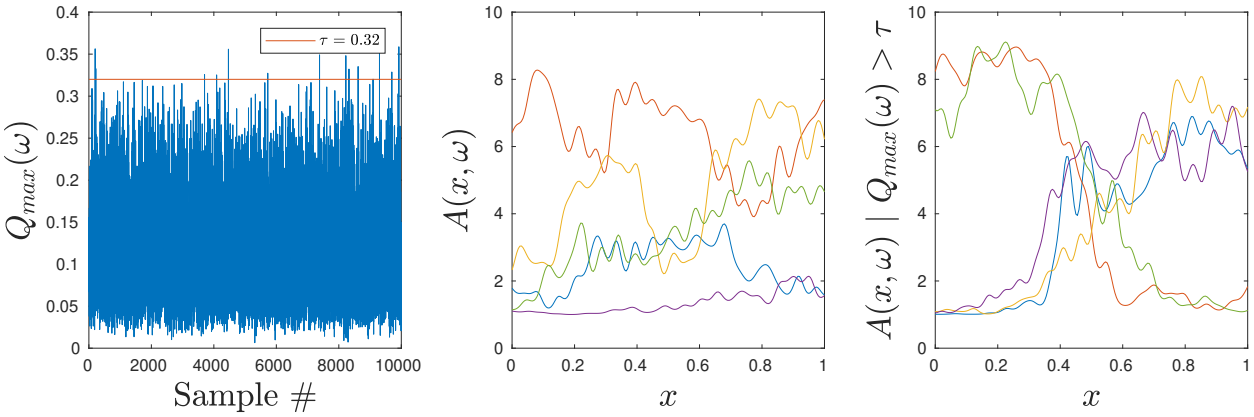

Figure 1: Samples of $Q_{max}(\omega)$ (left panel), $A(x,\omega)$ (middle panel), $A(x,\omega)\,|\,Q_{max}(\omega) > \tau$ for $\tau = 0.32$ (right panel).

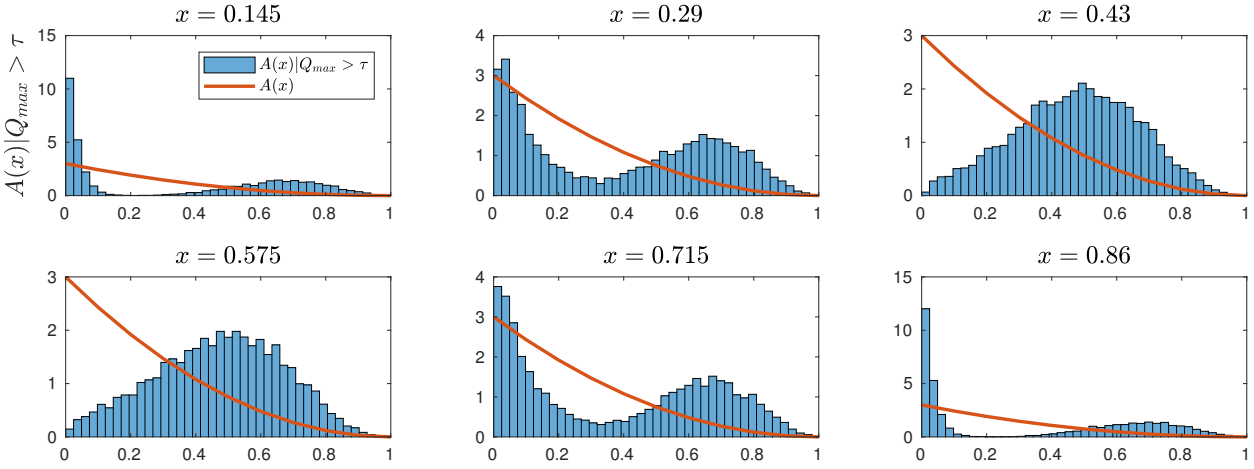

Figure 2: Histogram of $A(x,\omega)|Q_{max}(\omega) > \tau$ for $\tau = 0.32$ using 10000 samples compared to the pdf of $A(x,\omega)$ for a few values of $x \in [0,1]$. The range of $A(x,\omega)$ is scaled down to $[0,1]$.

We now seek indicators that aid in predicting if $Q_{max} > \tau$. Ideally, they should not resemble (2.1) since analytical solutions are unavailable in practice. But to facilitate our search, we work backwards and assume that (2.1) is known. From (2.2), Q_{max} can be computed by first finding the critical point $x = x_c(\omega) \in [0,1]$ which maximizes or minimizes $d(x,\omega) := U(1,x,\omega) - u(1,x)$ for each $\omega \in \Omega$. Since $d(0,\omega) = d(1,\omega) = 0$, the critical point $x_c(\omega)$ satisfies $d'(x_c(\omega),\omega) = 0$. Thus,

$$\frac{1}{A(x_c(\omega),\omega)} = \int_0^1 \frac{1}{A(y,\omega)}\,dy, \quad \omega \in \Omega, \tag{2.3}$$

is a necessary condition for $x_c(\omega)$, i.e., $x = x_c(\omega)$ is such that $(A(x,\omega))^{-1}$ achieves its spatial average (over $x \in [0,1]$). Substituting (2.3) to (2.2) yields an alternative expression for $Q_{max}(\omega)$:

$$\begin{aligned} Q_{max}(\omega) = |d(x_c(\omega),\omega)| &= \left| \left(\int_0^{x_c(\omega)} \frac{1}{A(y,\omega)}\,dy \right) \Big/ \left(\frac{1}{A(x_c(\omega),\omega)} \right) - x_c(\omega) \right| \\ &= \left| \int_0^{x_c(\omega)} \left(\frac{A(x_c(\omega),\omega)}{A(y,\omega)} - 1 \right) dy \right|. \end{aligned} \tag{2.4}$$

Qualitatively, for $Q_{max}(\omega)$ to be large, $1/A(y,\omega)$ must be substantially different in magnitude compared to $1/A(x_c(\omega),\omega)$ over $y \in [0, x_c(\omega)]$. This is confirmed in Figure 3 which displays 3 samples of $1/A(y,\omega)$ with their corresponding spatial averages $1/A(x_c(\omega),\omega)$ and $Q_{max}(\omega)$ values.

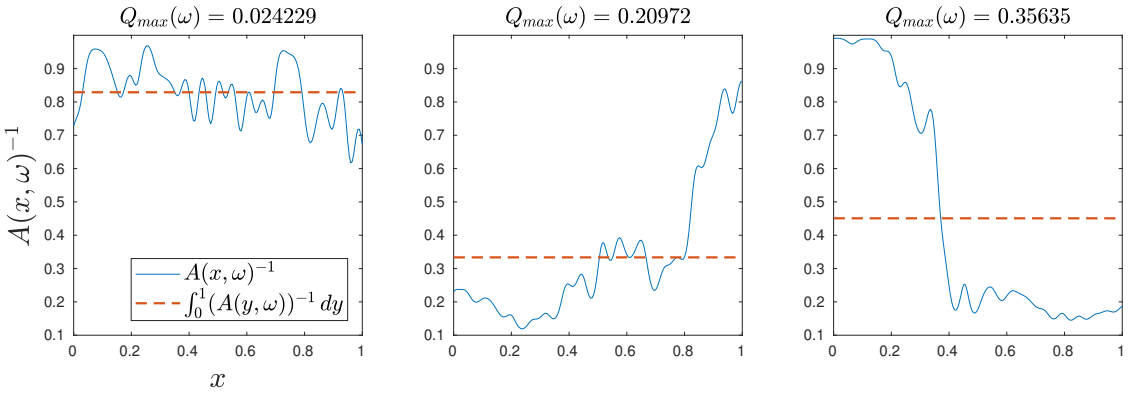

Figure 3: Samples of the inverse random field $A(x,\omega)^{-1}$ with their corresponding spatial averages and values of Q_{max}.

From Figure 3, it is speculated that two possible indicators for large Q_{max} are: the value of $1/A(x_c(\omega),\omega)$ and the frequency that $1/A(x,\omega)$ intersects its spatial average. We validate these indicators using data on $A(x,\omega)$ and $Q_{max}(\omega)$, momentarily neglecting the computational cost of obtaining such data.

Regarding the first indicator, it is observed that $1/A(x_c(\omega),\omega)$ cannot assume values close to 0 or 1 for $Q_{max}(\omega)$ to be large. Otherwise, if $1/A(x_c(\omega),\omega)$ were close to 0 or 1, the boundedness of $A(x,\omega)$ and (2.3) imply that $A(x,\omega)$ fluctuates near its spatial average. Hence, $\frac{A(x_c(\omega),\omega)}{A(y,\omega)} \approx 1$ resulting to a low $Q_{max}(\omega)$ according to (2.4). Probabilistic arguments also offer conclusions consistent with the above claims. Assume that $A(x,\omega)^{-1}$ is a homogeneous Gaussian process (despite its impossibility in our setting) on $x \in [0,1]$ such that $E[A(x,\omega)^{-1}] = \mu$ and $\mathrm{Cov}(A(x,\omega)^{-1}, A(y,\omega)^{-1}) = c(x,y)$ with $c(x,x) = \sigma^2$. From [9], $\int_0^1 A(y,\omega)^{-1}\,dy$ is a Gaussian random variable with mean μ, variance $\int_0^1\int_0^1 c(u,v)\,du\,dv$ and that the random vector $(A(x,\omega)^{-1}, \int_0^1 A(y,\omega)^{-1}\,dy)^T$ is jointly Gaussian with $Cov(A(x,\omega)^{-1}, \int_0^1 A(y,\omega)^{-1}\,dy) = \int_0^1 c(x,y)\,dy$ for all $x \in [0,1]$. In summary,

$$\begin{bmatrix} A(x,\omega)^{-1} \\ \int_0^1 A(y,\omega)^{-1}\,dy \end{bmatrix} \sim N\left(\begin{bmatrix} \mu \\ \mu \end{bmatrix}, \begin{bmatrix} \sigma^2 & \int_0^1 c(x,y)\,dy \\ \int_0^1 c(x,y)\,dy & \int_0^1\int_0^1 c(u,v)\,du\,dv \end{bmatrix} \right)$$

from which we infer

$$A(x,\omega)^{-1}\Big|\int_0^1 A(y,\omega)^{-1}\,dy=\xi\sim N\left(\mu+\frac{\int_0^1 c(x,y)\,dy}{\int_0^1\int_0^1 c(u,v)\,du\,dv}(\xi-\mu),\sigma^2-\frac{[\int_0^1 c(x,y)\,dy]^2}{\int_0^1\int_0^1 c(u,v)\,du\,dv}\right) \tag{2.5}$$

using properties of conditioning on Gaussian distributions. Three conclusions arise from (2.5): for a fixed $x\in[0,1]$,

1. $E[A(x,\omega)^{-1}|\int_0^1 A(y,\omega)^{-1}\,dy=\xi]>\mu$ if $\xi>\mu$,
2. $E[A(x,\omega)^{-1}|\int_0^1 A(y,\omega)^{-1}\,dy=\xi]<\mu$ if $\xi<\mu$,
3. $Var[A(x,\omega)^{-1}|\int_0^1 A(y,\omega)^{-1}\,dy=\xi]<\sigma^2$.

The first and second state that for all $x\in[0,1]$, if the spatial average of the random field is large (or small), the values of the random field also tend to be large (or small). Note that for each x, $A(x,\omega)^{-1}$ and $\int_0^1 A(y,\omega)^{-1}\,dy$ are positively correlated since in our example $\mathrm{Cov}(A(x,\omega)^{-1},\int_0^1 A(y,\omega)^{-1}\,dy)=\int_0^1 c(x,y)\,dy>0$. Meanwhile, the third suggests that the variance of the random field decreases if it is known that its spatial average assumes some value. These conclusions are consistent with our discussion earlier that for large (or small) $\int_0^1 A(y,\omega)^{-1}\,dy$, values of $A(x,\omega)^{-1}$ fluctuate more closely near $\int_0^1 A(y,\omega)^{-1}\,dy$ on average leading to low $Q_{max}(\omega)$. Hence, values of the indicator $1/A(x_c(\omega),\omega)$ cannot be close to 0 or 1 for large $Q_{max}(\omega)$. Data confirms this as shown in the left panel of Figure 4 which displays 10000 samples of $(Q_{max}(\omega),\int_0^1 A(y,\omega)^{-1}\,dy)$ and 5000 samples of $(Q_{max}(\omega),\int_0^1 A(y,\omega)^{-1}\,dy\,|\,Q_{max}(\omega)>\tau)$. We deduce that a necessary condition for $Q_{max}(\omega)>\tau$ is that $1/A(x_c(\omega),\omega)\in[0.15,0.6]$ approximately.

Under a similar reasoning, we intuit that a second indicator for $Q_{max}(\omega)>\tau$ is that the number of times $n_{x_c}(\omega)$ that $A(x,\omega)^{-1}$ crosses its spatial average cannot be substantial, i.e. the cardinality of the set $\{x_c(\omega)\,|\,\frac{1}{A(x_c(\omega),\omega)}=\int_0^1\frac{1}{A(y,\omega)}\,dy\}$ is small. Notice that the first indicator does not necessarily imply the latter and vice versa. Data presented in the middle panel of Figure 4 supports this indicator in which 10000 samples of $(Q_{max}(\omega),n_{x_c}(\omega))$ and 5000 samples of $(Q_{max}(\omega),n_{x_c}(\omega)\,|\,Q_{max}(\omega)>\tau)$ are plotted. The plot suggests that a necessary condition for $Q_{max}(\omega)>\tau$ is that $n_{x_c}(\omega)\le 5$.

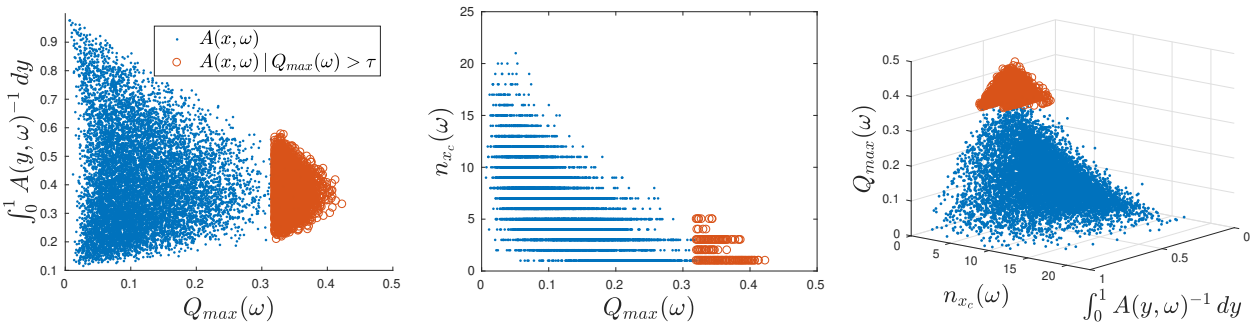

Figure 4: Samples of $(Q_{max}(\omega),\int_0^1 A(y,\omega)^{-1}\,dy)$ (left panel), $(Q_{max}(\omega),n_{x_c}(\omega))$ (middle panel), and $(\int_0^1 A(y,\omega)^{-1}\,dy,n_{x_c}(\omega),Q_{max}(\omega))$ (right panel). Conditional samples corresponding to $Q_{max}(\omega)>\tau$ are indicated by unfilled orange circles in each panel.

The above discussion is summarized in the right panel of Figure 4 which exhibits 10000 samples of $(\int_0^1 A(y,\omega)^{-1}\,dy,n_{x_c}(\omega),Q_{max}(\omega))$ and 5000 samples of $(\int_0^1 A(y,\omega)^{-1}\,dy\,|\,Q_{max}(\omega)>\tau,n_{x_c}(\omega)\,|\,Q_{max}(\omega)>\tau,Q_{max}(\omega))$. We have therefore identified 2 indicators which serve as necessary conditions that aid in predicting if Q_{max} is large. However, these indicators were deduced assuming that the analytical solution (2.1) were known which might not be the case in practice. We show in the next section that the search for indicators, assuming they exist, can be facilitated in the case when $A(x,\omega)$ is represented as a finite-dimensional noise model.

2.2 Finite-dimensional noise model

We revisit Example 1 but examine an alternative expression for the random field $A(x,\omega)$.

Example 2 *Consider the identical setup outlined in Example 1. Recast the random field $A(x,\omega)$ into a parametric form $A(x,\omega) = A(x,Z(\omega)) = a_0 + \sum_{k=1}^{\infty} Z_k(\omega)a_k(x)$ where $Z(\omega) = (Z_1(\omega), Z_2(\omega), \dots)$, $\{Z_k(\omega)\}_{k=1}^{\infty}$ are possibly non-Gaussian dependent random variables such that $E[Z_k] = 0\ \forall k$, and $\{a_k(x)\}_{k=1}^{\infty}$ are deterministic basis functions. The quantity of interest (2.2) can be rewritten as $Q_{max}(\omega) = Q_{max}(Z(\omega)) = \max_{x\in[0,1]} |U(1,x,Z(\omega)) - u(1,x)|$. The objective is to seek indicators based on components of $Z(\omega)$ that signal large $Q_{max}(Z(\omega))$.*

An approach to construct such a parametric form $A(x,Z(\omega))$ for the random field $A(x,\omega)$ is through the Karhunen-Loève (KL) expansion [10]. The method in [11] for performing the KL expansion via the singular-value decomposition (SVD) is adopted in view of numerical stability. For purposes of numerical implementation, M samples $\omega = \omega_1, \dots, \omega_M \in \Omega$ of $A(x,\omega)$ are discretized at spatial locations $x_k = k\Delta x, k = 0, \dots, N-1, \Delta x = \frac{1}{N-1}$, to form the $M \times N$ matrix $\mathbf{A}$ whose (j,k)-entry is $A(x_k,\omega_j)$, the value of the jth simulated sample of $A(x,\omega)$ at $x = x_k$. If $I_{M\times N}$ is the $M \times N$ identity matrix, the SVD is applied to $\mathbf{A} - a_0 I_{M\times N}$ to obtain M samples of $Z_k(\omega)$ and values of $\{a_k(x_k)\}_{k=1}^{N}$. Due to this approximation, the parametric form of $A(x,\omega)$ is written as a finite sum

$$A(x,\omega) = A(x,Z(\omega)) = a_0 + \sum_{k=1}^{N} Z_k(\omega)a_k(x),\ \ Z(\omega) = (Z_1(\omega), \dots, Z_N(\omega)),\ x \in [0,1]. \tag{2.6}$$

Since $A(x,\omega) \in [\alpha,\beta]$ in Example 1, $\{Z_k\}_{k=1}^{N}$ are bounded random variables and hence the computer implementation of $A(x,Z(\omega))$ in (2.6) satisfies $A(x,Z(\omega)) \in [\gamma_1,\gamma_2]$ where γ_1,γ_2 approach α,β as $N \to \infty$. In the simulations below, we adhere to the representation (2.6) with $\Delta x = 0.005$.

Using the same parameter values in Example 1, the top 4 panels of Figure 5 plot the first four interpolated basis functions $a_k(x)$ while the bottom 4 panels contain histograms of $\{Z_k(\omega)\}_{k=1}^{4}$ which correspond to the random variables with the largest variance. These plots are obtained from 10000 samples of $A(x,\omega)$. In addition, each of the bottom panels show 14 asterisks which represent samples of $Z_k | Q_{max} \geq \tau$ for $k = 1, \dots, 4$. Without making conclusions from this low sample size, it is observed that samples of Z_k for which Q_{max} is large are clustered. This behavior is not surprising as the calculations below illustrate.

Property 1 *Let $\hat{z}, \widetilde{z} \in \mathbb{R}^N$ be samples of $Z(\omega)$. It follows that $|Q_{max}(\hat{z}) - Q_{max}(\widetilde{z})| \leq \max_{x\in[0,1]} |U(1,x,\hat{z}) - U(1,x,\widetilde{z})|$.*

To show this, first note that $|\max_{x\in[0,1]} f(x) - \max_{x\in[0,1]} g(x)| \leq \max_{x\in[0,1]} |f(x) - g(x)|$ for any function f, g. This follows since $\max_{x\in[0,1]} f(x) \leq \max_{x\in[0,1]} (|f(x) - g(x)| + g(x)) \leq \max_{x\in[0,1]} |f(x) - g(x)| + \max_{x\in[0,1]} g(x)$. Likewise, $\max_{x\in[0,1]} g(x) \leq \max_{x\in[0,1]} |f(x) - g(x)| + \max_{x\in[0,1]} f(x)$.

Hence,

$$\begin{aligned} |Q_{max}(\hat{z}) - Q_{max}(\widetilde{z})| &\leq \max_{x\in[0,1]} \Big| |U(1,x,\hat{z}) - u(1,x)| - |U(1,x,\widetilde{z}) - u(1,x)| \Big| \\ &\leq \max_{x\in[0,1]} \Big| (U(1,x,\hat{z}) - u(1,x)) - (U(1,x,\widetilde{z}) - u(1,x)) \Big|. \end{aligned}$$

Property 2 *Recall that $A(x,\omega) \in [\gamma_1,\gamma_2]$ a.s. so that $1/A(x,\omega) \in [1/\gamma_2, 1/\gamma_1]$ a.s. For $\hat{z}, \widetilde{z} \in \mathbb{R}^N$, there exists $C > 0$ such that $|Q_{max}(\hat{z}) - Q_{max}(\widetilde{z})| \leq C\|A(x,\hat{z}) - A(x,\widetilde{z})\|_{L^2[0,1]}$.*

Since $U(1,x,Z) = \left(\int_0^x \frac{dy}{A(y,Z)}\right)\left(\int_0^1 \frac{dy}{A(y,Z)}\right)^{-1}$, let $\hat{I}(x) = \int_0^x \frac{dy}{A(y,\hat{z})}$ and $\widetilde{I}(x) = \int_0^x \frac{dy}{A(y,\widetilde{z})}$ so that

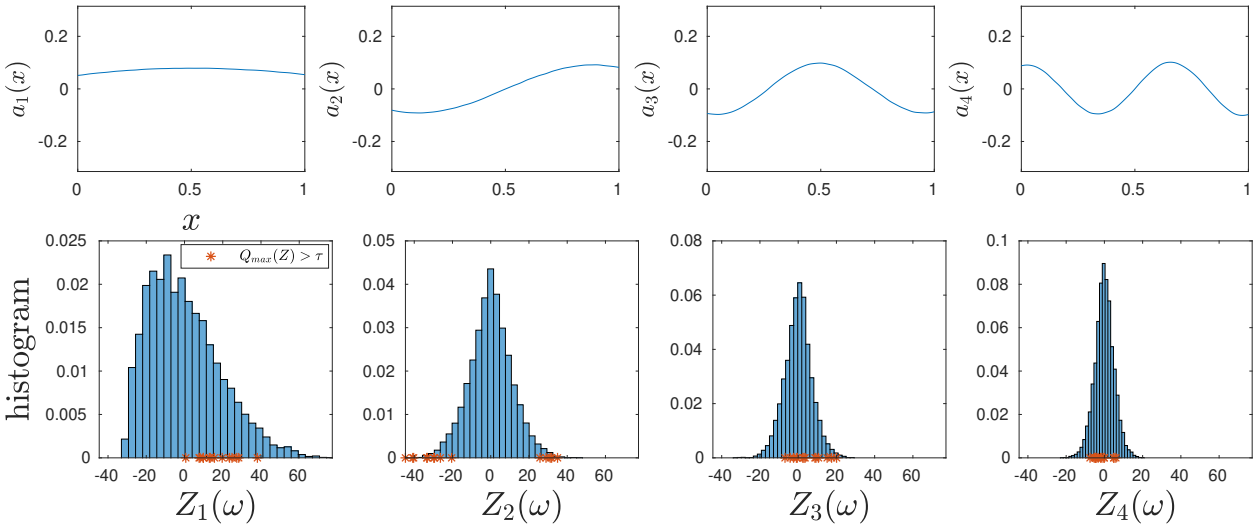

Figure 5: Top panels: Interpolated values of the basis functions $\{a_k(x)\}_{k=1}^4$. Bottom panels: Histogram of 10000 samples of $\{Z_k\}_{k=1}^4$ with 14 samples of $Z_k \,|\, Q_{max} > \tau$ for $\tau = 0.32$ in each subplot marked by asterisks.

$U(1, x, \hat{z}) = \hat{I}(x)/\hat{I}(1)$ and $U(1, x, \widetilde{z}) = \widetilde{I}(x)/\widetilde{I}(1)$. From [12, p. 194],

$$|U(1, x, \hat{z}) - U(1, x, \widetilde{z})| \leq \frac{\widetilde{I}(1)|\hat{I}(x) - \widetilde{I}(x)| + \widetilde{I}(x)|\widetilde{I}(1) - \hat{I}(1)|}{\hat{I}(1)\widetilde{I}(1)}. \tag{2.7}$$

We note that $1/\gamma_2 \leq \hat{I}(x), \widetilde{I}(x) \leq 1/\gamma_1$ and that

$$\begin{aligned} |\hat{I}(x) - \widetilde{I}(x)| \leq \frac{1}{\gamma_1^2} \int_0^x |A(y, \widetilde{z}) - A(y, \hat{z})| \, dy &\leq \frac{1}{\gamma_1^2} \int_0^1 |A(y, \widetilde{z}) - A(y, \hat{z})| \, dy \\ &\leq \frac{1}{\gamma_1^2} \|A(x, \hat{z}) - A(x, \widetilde{z})\|_{L^2[0,1]} \end{aligned} \tag{2.8}$$

for all $x \in [0, 1]$ by the Cauchy-Schwarz inequality. Property 2 then follows from Property 1 and (2.7), (2.8).

Property 3 *Suppose that the parametric form in* (2.6) *is constructed with* $a_k(x)$ *appropriately scaled so that* $\int_0^1 a_k^2(x) \, dx = 1$ *for all* k. *For* $\hat{z}, \widetilde{z} \in \mathbb{R}^N$, *there exists* $C > 0$ *such that* $|Q_{max}(\hat{z}) - Q_{max}(\widetilde{z})| \leq C \left[\sum_{k=1}^N |\hat{z}_k - \widetilde{z}_k|^2 \right]^{1/2}$.

This is established using Property 2 and the fact that $\int_0^1 a_j(x) a_k(x) \, dx = \delta_{jk}$ for $j, k = 1, \ldots, N$ where δ_{jk} is the Kronecker delta function. Similar results can be derived for other differential equations for which results of the form $\|U(t, x, \hat{z}) - U(t, x, \widetilde{z})\| \leq C\|A(x, \hat{z}) - A(x, \widetilde{z})\|$ hold under suitable norms.

Property 3 explains the clustering behavior of the samples of $Z_k \,|\, Q_{max} > \tau$ in the bottom panel of Figure 5. In particular, the range of $Z_2 \,|\, Q_{max} > \tau$ is a union of 2 disjoint sets. This is supported by Figure 6 which illustrates the pdf of Q_{max} conditioned on Z_2 obtained using kernel density estimation based on 250000 samples of (Z_2, Q_{max}). A consequence of this is that the failure domain $F = \{z \in \mathbb{R}^N \,|\, Q_{max}(z) > \tau\}$ is also a union of two disjoint sets (i.e. connected components). Determining whether $Z \in F$ requires characterizing the geometry of F which is challenging in high dimensions. This is why indicators based on components of Z are devised to aid in predicting if $Q_{max} > \tau$. From Figures 5, 6, the simplest example of an indicator

could be the value that Z_2 takes, i.e. $|Z_2| > \rho$ where ρ is calibrated, which can be thought of as intimately related to the second indicator n_{x_c} in Section 2.1.

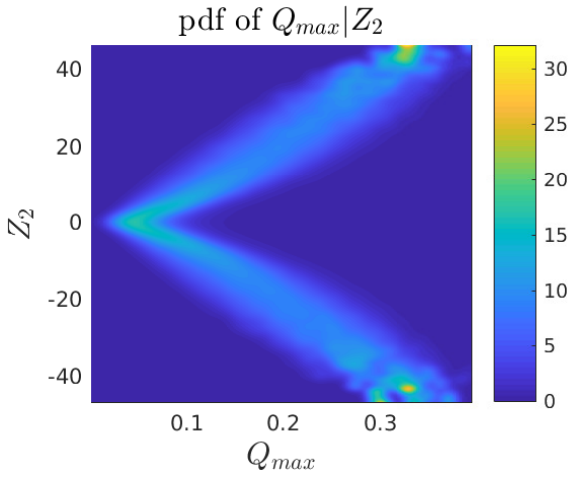

Figure 6: Pdf of Q_{max} conditioned on Z_2 constructed using kernel density estimation based on 250000 samples of (Z_2, Q_{max}).

We also remark that if the parametric form (2.6) were constructed for $A(x,\omega)^{-1}$ instead of $A(x,\omega)$, then the random variable Z_1 associated with the basis function $a_1(x)$ in the resulting expansion can be another indicator that is closely related to the first indicator $\int_0^1 A(y,\omega)^{-1}\,dy$ introduced in Section 2.1. In summary, Section 2 highlights that indicators in certain settings may be found that act as classifiers to signal large Q_{max}. However, the discussion above also presents challenges associated with this approach. Without any physical intuition or access to any analytical formulas, identifying indicators can be difficult and they strongly depend on the problem context. For example, the analysis in Example 1 has to be repeated if the quantity of interest is changed. Even if potential indicators can be identified, one needs to 1) verify that they correlate with large quantities of interest and 2) quantify their range of values for prediction. Doing so requires a large number of data on rare events for which simulation can be costly. In the following section, we address these issues via a multifidelity approximation to Q_{max} which is more directly applicable across various contexts.

3 Multifidelity physics-based surrogate models

Motivated by the computational cost of validating and calibrating indicators and their non-universal nature, we construct a multifidelity approximation to the QoI which offers another scheme for classification. Section 3.1 describes the surrogate model we use while the multifidelity approach is elaborated in Section 3.2. The capacity of each method as a classifier is assessed and their advantages and disadvantages are outlined.

In the remainder of this work, we solve a finite-dimensional version of (1.1), i.e. the stochastic equation

$$\frac{\partial^2 U(t,x,Z(\omega))}{\partial t^2} = \frac{\partial}{\partial x}\left(A(x,Z(\omega))\frac{\partial}{\partial x}U(t,x,Z(\omega))\right), \qquad x\in[0,1],\, t>0,\, \omega\in\Omega$$

with initial and boundary conditions as in (1.2) where $A(x,Z(\omega)) = a_0 + \sum_{k=1}^{N} Z_k(\omega)a_k(x)$, $Z(\omega) = (Z_1(\omega),\ldots,Z_N(\omega))$ is a parametric random field. The reference will therefore be the finite-dimensional solution $U(t,x,Z)$ from which quantities of interest are computed. Surrogate models for numerically solving stochastic equations require that the probability space be discretized, i.e. replace $A(x,\omega)$ in (1.1) with $A(x,Z(\omega))$. However, in the subsequent analysis and in our proposed framework, it is not assumed that the original input random

field is infinite-dimensional which is then truncated to obtain a parametric random field. Rather, only parametric random fields are considered. The former introduces an extra layer of approximation in addition to the approximation induced by using a surrogate model which we do not address here.

3.1 SROM-based surrogate model

The most straightforward approach to identify the samples of interest is to evaluate the full model $Q_{max}(Z)$. Since $Q_{max}(Z) > \tau$ is a rare event, obtaining n samples of $A(x,Z) \,|\, Q_{max}(Z) > \tau$ to understand why rare events occur requires at least $n/P(Q_{max}(Z) > \tau)$ full model solves which can be prohibitively costly. It is therefore reasonable to simulate samples of $A(x,\omega) \,|\, \widetilde{Q}_{max}(Z) > \tau$ instead where $\widetilde{Q}_{max}(Z)$ is a surrogate model that converges almost surely to $Q_{max}(Z)$. Otherwise, a surrogate model which converges to $Q_{max}(Z)$ in $L^p(\Omega)$ for p large can be a suitable approximation since if $f \in L^\infty(\Omega)$, $\lim_{p\to\infty} \|f\|_{L^p(\Omega)} = \|f\|_{L^\infty(\Omega)}$. Once the surrogate is constructed, it functions as a classifier that not only indicates if $Q_{max} > \tau$ or otherwise for a sample $A(x,Z)$ but also provides an approximate value of Q_{max}.

We remark that the mode of convergence of the surrogate is crucial for this objective. For instance, it is possible for $P(\widetilde{Q}_{max} > \tau) \approx P(Q_{max} > \tau)$ yet the samples of $A(x,Z) \,|\, \widetilde{Q}_{max} > \tau$ and $A(x,Z) \,|\, Q_{max} > \tau$ may greatly differ. We demonstrate this for a 1-dimensional input with simple QoI in Example 3 in which the polynomial chaos (PC) expansion [13, 14], which converges in $L^2(\Omega)$ but not almost surely, is used as a surrogate.

Example 3 *Suppose that the response is given by $g(W) = \Phi(W)$, $W \sim N(0,1)$. The objective is to compare the sets of samples $W \,|\, g(W) > \tau$ and $W \,|\, g^{PCE}(W) > \tau$ for specified τ where $g^{PCE}(W)$ is the PC expansion of $g(W)$.*

Since $g(W) \sim U(0,1)$, $E[g(W)^2] < \infty$ and $g(W)$ admits the representation $g^{PCE}(W) = \beta_0 + \sum_{k=0}^{n} \beta_{2k+1} h_{2k+1}(W)$ where $h_j(W)$ are Hermite polynomials orthogonal with respect to the standard normal pdf while $\beta_0 = \frac{1}{2}, \beta_{2k} = 0, k = 1,\dots,n$ and $\beta_{2k+1} = (-1)^k \frac{(2k)!}{2^{2k+1}\sqrt{\pi}(2k+1)!k!}, k = 0,1,\dots,n$. We choose $\tau = 0.9825$ so that $P(g(W) > \tau) = 0.0175$. In practice, the truncation level for the PC expansion must be kept low to prevent an explosion in the number of terms if W is high-dimensional. In our simulations, we select $n = 5$ so that $P(g^{PCE}(W) > \tau) \approx 0.0190$ which has relative error 8.57%. The Monte Carlo estimates for the mean and variance are $E[g^{PCE}(W)] \approx 0.4990, Var(g^{PCE}(W)) \approx 0.0830$ which are consistent with that of a uniform distribution. Figure 7 shows histograms of 10252 samples of $W \,|\, g(W) > \tau$ and 10094 samples of $W \,|\, g^{PCE}(W) > \tau$, respectively. It is seen that the histogram due to the PCE surrogate is biased and has a thinner tail compared to the true histogram. For example, $P(W > 2.5 \,|\, g(W) > \tau) \approx 0.3641$ whereas $P(W > 2.5 \,|\, g^{PCE}(W) > \tau) \approx 0.0148$.

Example 3 affirms why the surrogate models for reliability engineering are insufficient for our objective. Since $g^{PCE}(W)$ is able to match the second moment properties of $g(W)$, it suggests that $g^{PCE}(W)$ is a good approximation to $g(W)$ in the high probability region of W. Furthermore, from the calculation above, $P(g^{PCE}(W) > \tau) \approx P(g(W) > \tau)$. Despite the approximation properties of $g^{PCE}(W)$, the samples $W \,|\, g^{PCE}(W) > \tau$ of W that cause large response differ substantially from $W \,|\, g(W) > \tau$. We therefore strive to design a surrogate so that these two sets of samples closely approximate each other.

In view of these issues, an SROM-based surrogate model [15] $\widetilde{U}(t,x,Z)$ for the full model $U(t,x,Z)$ is employed from which the approximation for $Q_{max}(Z)$ arises. The surrogate is constructed by partitioning the range $Z(\Omega) = \Gamma \subset \mathbb{R}^N$ of Z using Voronoi cells $\{\Gamma_k\}_{k=1}^M$ whose centers $\{z_k\}_{k=1}^M \subset \mathbb{R}^N$ are samples of Z chosen to approximate its distribution. A first-order Taylor expansion at $Z = z_k$ for every partition is computed to obtain

$$\widetilde{U}(t,x,Z) = \sum_{k=1}^{M} (U(t,x,z_k) + \nabla_Z U(t,x,z_k) \cdot (Z - z_k)) \mathbb{1}_{Z \in \Gamma_k} \tag{3.1}$$

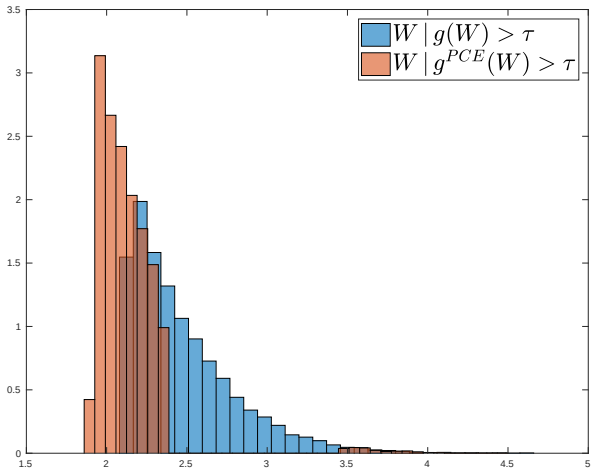

Figure 7: Histograms of 10210 samples of $W \mid g(W) > \tau$ and 10317 samples of $W \mid g^{PCE}(W) > \tau$ for Example 3.

from which the surrogate approximation $\widetilde{Q}_{max}(Z) = \max_{x\in[0,1]} |\widetilde{U}(1,x,Z) - u(1,x)|$ is then derived. Under mild conditions, it can be shown that $\widetilde{U}(t,x,Z) \to U(t,x,Z)$ almost surely and in $L^p(\Omega)$ as $M \to \infty$. From Property 1, these convergence results also hold for $\widetilde{Q}_{max}(Z)$.

To improve the performance of (3.1), in previous work [16], we proposed an adaptive construction of (3.1) that sequentially refines the surrogate $\widetilde{U}(t,x,Z)$ to enhance its accuracy in high-probability regions of Γ and regions for which the discrepancy between the QoI $Q(Z)$ and the surrogate QoI $\widetilde{Q}(Z)$ is large. This was achieved by setting the refinement criterion as $\|Q(Z) - \widetilde{Q}(Z)\|_{L^p(\Omega)}$ for specified p which also served as the stopping criterion. The L^p error was crudely approximated via $\dfrac{1}{n_1 + \cdots + n_M} \displaystyle\sum_{k=1}^{M} \sum_{\ell=1}^{n_k} |Q(z_k^\ell) - \widetilde{Q}(z_k^\ell)|^p$ where $\{z_k^\ell\}_{\ell=1}^{n_k} \subset \Gamma_k$ are a few samples of Z at which we evaluate the response and from which we select the next point at which the gradient in (3.1) is calculated. Evaluating the response is less expensive than computing multiple partial derivatives and furthermore, this approximation ensures that the L^p error is estimated via contributions from each partition. As the adaptive algorithm progresses, Γ_k decreases in size and the L^p error in this partition can be approximated using a few samples only.

We now investigate the use of the SROM-based surrogate for $Q_{max}(Z)$ to produce samples that resemble $A(x,Z) \mid Q_{max}(Z) > \tau$. Two surrogate models are examined: denote by $\widetilde{Q}_{max}^{direct}$ the direct construction (3.1) and by $\widetilde{Q}_{max}^{adapt}$ the adaptive construction described above.

Example 4 *We revisit the problem setup in Example 1 where it is assumed instead that the true input random field is parametric expressed as*

$$A(x, Z(\omega)) = a_0 + \sum_{k=1}^{10} Z_k(\omega) a_k(x), \quad Z(\omega) = (Z_1(\omega), \ldots, Z_{10}(\omega)), \quad x \in [0,1], \tag{3.2}$$

where $\{a_k\}_{k=0}^{10}, \{Z_k\}_{k=1}^{10}$ *are exactly the same basis functions and random variables appearing in* (3.2)*. The performance of* $\widetilde{Q}_{max}^{direct}$ *and* $\widetilde{Q}_{max}^{adapt}$ *in generating conditional samples that approximate* $A(x,Z) \mid Q_{max}(Z) > \tau$ *is compared with the computational cost of both surrogates being similar. Their use in identifying samples of* $A(x,Z)$ *that yield large* Q_{max} *is also examined.*

This example is not a continuation of Examples 1, 2 in which the input random field is initially infinite-dimensional that is truncated to 10 basis functions. Any random field that depends on $Z(\omega)$ can be used in

lieu of (3.2). Nevertheless, we have chosen to select the first 10 basis functions in (2.6) for Example 4 as it sufficiently approximates (2.6) and numerically guarantees that $A(x, Z(\omega))$ in (3.2) satisfies $A(x, Z(\omega)) > 0$. In particular, (3.2) includes the basis function $a_2(x)$ which is primarily responsible for large values of the quantity of interest, cf. Figure 6.

In the calculations that follow, we adjust $\tau = 0.3172$ so that $P(Q_{max}(Z) > \tau) \approx 0.00224$ because we are now working with the truncated random field (3.2) instead of (2.6). It is assumed that calculating a partial derivative in the gradient term in (3.1) is as costly as evaluating the response $U(t, x, Z)$. We computed the gradients from (2.1) but they can also be numerically estimated by differentiating (1.1) and solving the resulting differential equations satisfied by $\nabla_Z U(t, x, Z)$ as in [16]. The surrogate $\widetilde{Q}_{max}^{direct}$ obtained comprises 370 Voronoi cells and constitutes 370 response evaluations and 370×10 gradient evaluations for a total of 4070 computational units. In contrast, $\widetilde{Q}_{max}^{adapt}$ is composed of 325 Voronoi cells which implies 3575 gradient and response calculations and an additional 465 response evaluations that are used to estimate the refinement criterion at each iteration, resulting in a total of 4040 computational units. For the adaptive construction, we used $p = \infty$ for the refinement criterion since we are targeting rare events and employed the global sampling and neighbor-based refinement configuration, following terminology in [16].

Figure 8 compares the histograms of $Q_{max}, \widetilde{Q}_{max}^{adapt}$ and $\widetilde{Q}_{max}^{direct}$ constructed using 10000 samples. The probability of failure estimates for each surrogate are $P(\widetilde{Q}_{max}^{adapt} > \tau) \approx 0.001834$ and $P(\widetilde{Q}_{max}^{direct} > \tau) \approx 0.001754$ obtained using 500000 Monte Carlo samples of Z. The discrepancy between the 2 surrogate models is not obvious unless we examine samples of $Z \,|\, Q_{max}(Z) > \tau$ which is relevant to our objectives. Figures 9 and 10 contrast the histograms of $Z_i \,|\, Q_{max}(Z)$ with $Z_i \,|\, \widetilde{Q}_{max}^{adapt}(Z)$ and $Z_i \,|\, \widetilde{Q}_{max}^{direct}$, respectively, for $i = 1, \ldots, 6$. These histograms result from 10000 samples of $Z \,|\, Q_{max}(Z)$, 10001 samples of $Z \,|\, \widetilde{Q}_{max}^{adapt}(Z)$ and 10022 samples of $Z \,|\, \widetilde{Q}_{max}^{direct}(Z)$. Based on these figures, the advantages of using an adaptive construction is now evident since the histograms of $Z_1 \,|\, \widetilde{Q}_{max}^{direct}$ and $Z_3 \,|\, \widetilde{Q}_{max}^{direct}$ appear biased while the modes of $Z_2 \,|\, \widetilde{Q}_{max}^{direct}$ do not appear to have the same frequency.

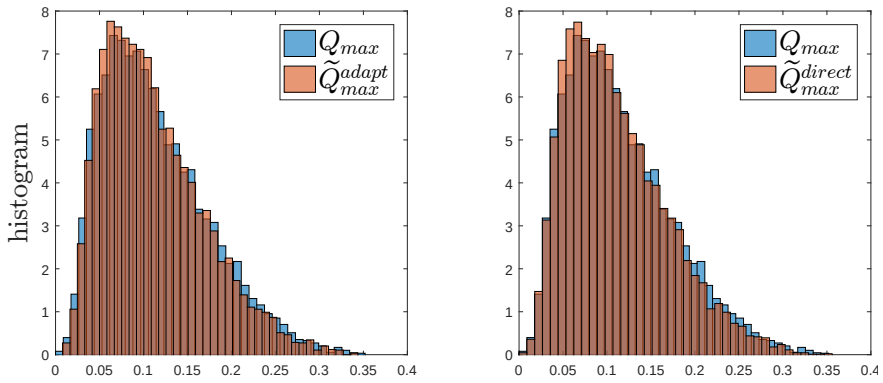

Figure 8: Comparison of the histogram of Q_{max} vs $\widetilde{Q}_{max}^{adapt}$ (left panel) and Q_{max} vs $\widetilde{Q}_{max}^{direct}$ (right panel) using 10000 samples.

Because of the performance of $\widetilde{Q}_{max}^{adapt}(Z)$, we further study its suitability in achieving our objectives. How well does $\widetilde{Q}_{max}^{adapt}(Z)$ classify whether $Q_{max}(Z) > \tau$ or $Q_{max}(Z) \leq \tau$ for a given Z? To accomplish this, tools and metrics commonly used in machine learning and pattern recognition [17] are adopted. They have also been invoked in [6, 7] to validate and calibrate indicators. These metrics can be extracted from a confusion matrix which tabulates in each entry the frequency that a classifier is consistent or not with the truth; see Table 1 for a visual definition where N_s is the total number of samples. In particular, we calculate precision and recall to assess the quality of $\widetilde{Q}_{max}^{adapt}(Z)$ as a classifier. Precision is defined as $P(Q_{max}(Z) > \tau \,|\, \widetilde{Q}_{max}^{adapt}(Z) > \tau)$, i.e. the fraction of $P(Q_{max}(Z) > \tau, \widetilde{Q}_{max}^{adapt}(Z) > \tau)$ (true positives) with respect to the sum of $P(Q_{max}(Z) > \tau, \widetilde{Q}_{max}^{adapt}(Z) > \tau)$ (true positives) and $P(Q_{max}(Z) \leq \tau, \widetilde{Q}_{max}^{adapt}(Z) > \tau)$ (false positives) while recall is defined as $P(\widetilde{Q}_{max}^{adapt}(Z) > \tau \,|\, Q_{max}(Z) > \tau)$, i.e. the fraction of $P(Q_{max}(Z) > \tau, \widetilde{Q}_{max}^{adapt}(Z) > \tau)$ (true positives) with respect to the sum of $P(Q_{max}(Z) > \tau, \widetilde{Q}_{max}^{adapt}(Z) > \tau)$ (true positives)

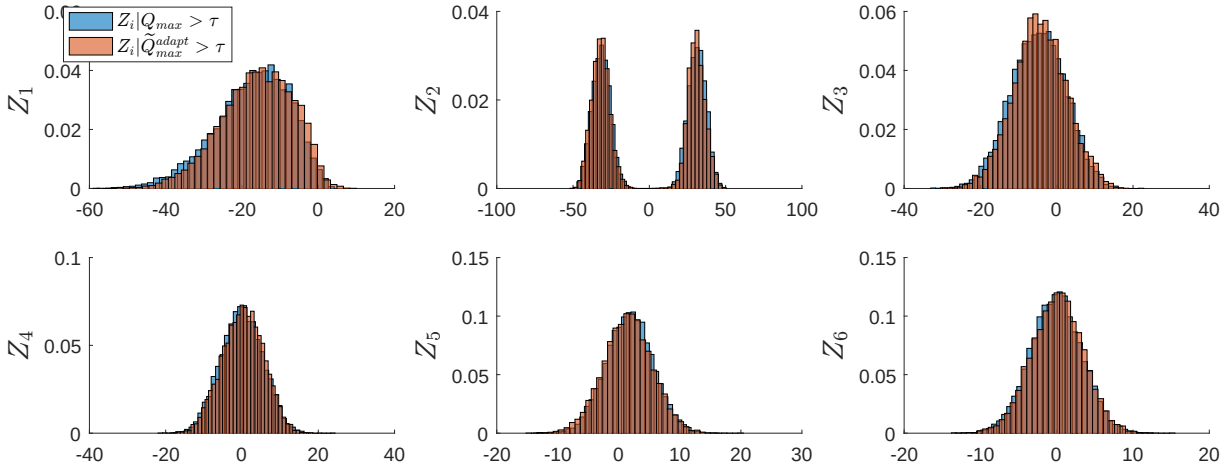

Figure 9: Comparison of histograms of $Z_i \,|\, Q_{max}(Z) > \tau$ and $Z_i \,|\, Q^{adapt}_{max}(Z) > \tau$ for $i = 1, \ldots, 6$.

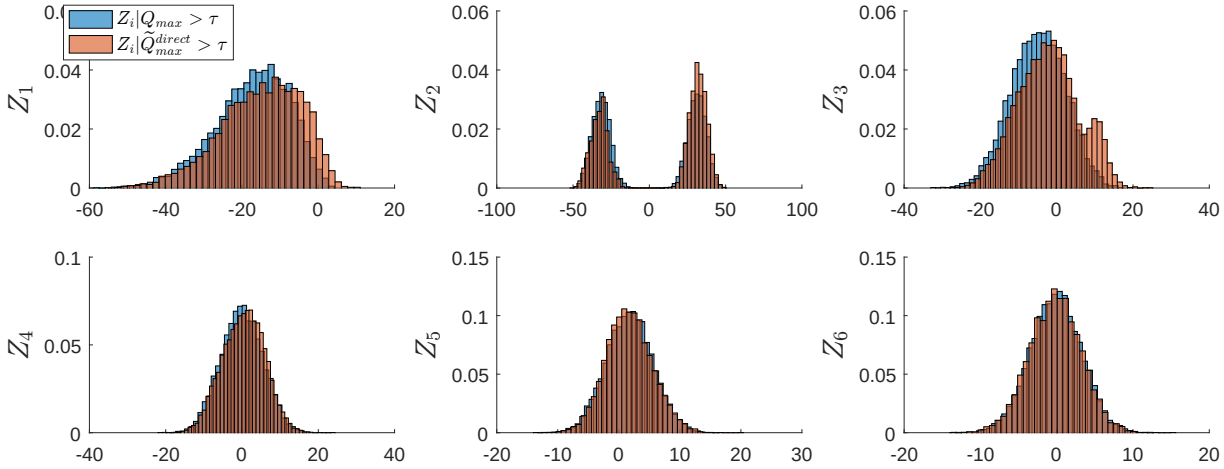

Figure 10: Comparison of histograms of $Z_i \,|\, Q_{max}(Z) > \tau$ and $Z_i \,|\, Q^{direct}_{max}(Z) > \tau$ for $i = 1, \ldots, 6$.

and $P(Q_{max}(Z) > \tau, \widetilde{Q}^{adapt}_{max}(Z) \leq \tau)$ (false negatives). In our setting, perfect precision implies that the failure domain of $\widetilde{Q}^{adapt}_{max}(Z)$ is a subset of the failure domain of $Q_{max}(Z)$ while the reverse is true for perfect recall. We also consider the failure rate which served as a basis for validation and calibration of indicators for extreme events in [7]. This is defined as the sum of the false omission rate and the false discovery rate or mathematically, $P(Q_{max}(Z) > \tau | \widetilde{Q}^{adapt}_{max}(Z) < \tau) + P(Q_{max}(Z) < \tau | \widetilde{Q}^{adapt}_{max}(Z) > \tau)$.

$\widetilde{Q}^{adapt}_{max}(Z)$ \ $Q_{max}(Z)$	Positive	Negative
Positive	$N_s P(Q_{max} > \tau, \widetilde{Q}^{adapt}_{max} > \tau)$	$N_s P(Q_{max} \leq \tau, \widetilde{Q}^{adapt}_{max} > \tau)$
Negative	$N_s P(Q_{max} > \tau, \widetilde{Q}^{adapt}_{max} \leq \tau)$	$N_s P(Q_{max} \leq \tau, \widetilde{Q}^{adapt}_{max} \leq \tau)$

Table 1: Definition of the confusion matrix.

$\widetilde{Q}_{max}^{adapt}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	11	1	12
Negative	7	9981	9988
Total	18	9982	10000

Table 2: Confusion matrix for $\widetilde{Q}_{max}^{adapt}(Z)$ based on 10000 samples of Z.

$\widetilde{Q}_{max}^{adapt}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	84	17	101
Negative	42	49857	49899
Total	126	49874	50000

Table 3: Confusion matrix for $\widetilde{Q}_{max}^{adapt}(Z)$ based on 50000 samples of Z.

Confusion matrix	Precision	Recall	Failure rate
Table 2	0.9167	0.6111	0.0840
Table 3	0.8317	0.6667	0.1691

Table 4: Precision, recall, and failure rate metrics computed from Tables 2, 3.

Tables 2 and 3 chart the confusion matrices of $\widetilde{Q}_{max}^{adapt}(Z)$ generated using 10000 and 50000 samples of Z, respectively, with the latter set of samples including the former. These sets of samples are referred to as the test data. The precision, recall, and failure rate metrics are summarized in Table 4 from the confusion matrices in Tables 2, 3. It is desirable for both precision and recall to be close to 1 and for the failure rate to be near 0. Consequently, the performance of $\widetilde{Q}_{max}^{adapt}(Z)$ as a classifier can still be improved. We also remark that using a small number of samples to construct the confusion matrix can result in unstable estimates of the frequency values. Table 5 shows the confusion matrix for $\widetilde{Q}_{max}^{adapt}(Z)$ using another set of 10000 samples of Z. The metrics based on Table 5 appear inflated compared to those from Table 3.

$\widetilde{Q}_{max}^{adapt}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	20	1	21
Negative	6	9973	9979
Total	26	9974	10000

Table 5: Confusion matrix for $\widetilde{Q}_{max}^{adapt}(Z)$ based on a different set of 10000 samples of Z.

More computational effort can certainly be expended to make $\widetilde{Q}_{max}^{adapt}(Z)$ more accurate, however, for the SROM-based surrogate, an additional Voronoi cell requires at least $N+1$ calculations for $Z \in \mathbb{R}^N$. Other surrogate models such as the sparse grid stochastic collocation also incur substantial cost when transitioning from one sparse grid level to another [16]. We deal with these issues by investigating the use of a multifidelity surrogate model as described in Section 3.2.

3.2 Multifidelity surrogate approach

A multifidelity surrogate leverages on various surrogate models with differing computational costs and levels of accuracy to accomplish an objective. The multifidelity approach we pursue consists of a single surrogate model and infrequent calls to the full model. Such approaches have already been applied in previous work [2, 3] in which the goal is computation of failure probabilities. Here, we adopt some ideas from these works to tackle our objectives. We first survey how these multifidelity surrogates were tailored to approximate the probability of failure $P(Q(Z) > \tau)$.

Let $\widetilde{Q}(Z)$ be the surrogate model to the QoI $Q(Z)$. Instead of the Monte Carlo estimate $\frac{1}{M}\sum_{i=1}^{M} \mathbb{1}_{\{\widetilde{Q}(z_i)>\tau\}}$ where $\{z_i\}_{i=1}^{M}$ are samples of Z, [2] advocates using estimate $\widetilde{P}_f^M = \frac{1}{M}\sum_{i=1}^{M} \mathbb{1}_{\{z_i \in \widetilde{F}\}}$ where

$$\widetilde{F} = \{z \,|\, \widetilde{Q}(z) > \tau + \gamma\} \cup \{z \,|\, |\widetilde{Q}(z) - \tau| < \gamma \text{ and } Q(z) > \tau\}$$

for specified $\gamma > 0$. In other words, if the value of the surrogate exceeds the threshold τ by a margin γ, it is believed that value of the full model is also above τ, while if the surrogate value is close to τ, the full

model is invoked to ascertain whether $Q(z) > \tau$ or $Q(z) \leq \tau$. Moreover, let $\widetilde{P}_f = P(Z \in \widetilde{F})$ be such that $\widetilde{P}_f^M$ is an estimator for $\widetilde{P}_f$ and let $p \geq 1$ such that $Q(Z), \widetilde{Q}(Z) \in L^p(\Omega)$. It was proven in [2] that for $\epsilon > 0$, $|P(Q(Z) > \tau) - \widetilde{P}_f| < \epsilon$ if $\gamma > \frac{1}{\epsilon^{1/p}} \|Q(Z) - \widetilde{Q}(Z)\|_{L^p(\Omega)}$.

In addition to [2], [3] tackled the same problem using a similar multifidelity approach. They introduced the concept of reliability which is based on an approximation to the error between $Q(Z)$ and $\widetilde{Q}(Z)$ and a constant called safety factor. Knowledge of this constant coupled with infrequent solves of the full model guarantees that their approach is able to exactly classify whether $Q(z) > \tau$ or $Q(z) \leq \tau$ for each sample z. The failure probability estimate from this method is hence identical to the Monte Carlo estimate obtained from the full model, i.e. $\frac{1}{M} \sum_{i=1}^{M} \mathbb{1}_{\{Q(z_i) > \tau\}}$.

Despite the demonstrated effectiveness of the abovementioned approaches in approximating failure probabilities, they may not be readily applied to our problem since convergence in probabilities may or may not be indicative of the samples. While the guarantee of [3] to exactly classify whether $Q(z) > \tau$ or $Q(z) \leq \tau$ at a reduced computational expense avoids this concern and accomplishes our objectives, the method is contingent on constants that need to be estimated. Likewise, the convergence bounds in [2] rely on unknown constants whereas the more practical approach the authors proposed is geared towards convergence in estimate of the failure probability.

We therefore modify the multifidelity surrogate in [2] to obtain $\widetilde{Q}^*(Z)$ defined in Algorithm 1. Succinctly, the full model is only invoked if $\widetilde{Q}(Z) > \tau$ to ascertain whether $Q(Z) > \tau$ or $Q(Z) \leq \tau$. Unlike [2], if $\widetilde{Q}(Z) > \tau + \gamma$ for $\gamma > 0$, Algorithm 1 does not assume that $Q(Z) > \tau$ nor does it invoke $Q(Z)$ if $\widetilde{Q}(Z) > \tau - \gamma$. We note the following analytical and practical properties of $\widetilde{Q}^*(Z)$:

1. It does not rely on constants to be specified except for the threshold value τ.
2. The frequency that the full model is invoked is $P(\widetilde{Q}(Z) > \tau)$ which can be inexpensively approximated using Monte Carlo simulation beforehand.
3. The proportion of false positives of $\widetilde{Q}^*(Z)$ acting as a classifier, i.e. $P(Q(Z) \leq \tau, \widetilde{Q}^*(Z) > \tau)$, is 0. This is because from Algorithm 1, $\{\widetilde{Q}^*(Z) > \tau\} = \{\widetilde{Q}(Z) > \tau\} \cap \{Q(Z) > \tau\}$ so that $\{\widetilde{Q}^*(Z) > \tau\} \cap \{Q(Z) \leq \tau\} = \emptyset$.
4. If Monte Carlo samples of $\widetilde{Q}^*(Z)$ are generated, the quality of the surrogate $\widetilde{Q}(Z)$ can be monitored during the sampling procedure by calculating $P(Q(Z) > \tau \,|\, \widetilde{Q}(Z) > \tau)$ using the samples produced thus far. If this estimate is zero, the surrogate needs to be refined further.

Bounds on the misclassification probability $P(Q(Z) > \tau, \widetilde{Q}^*(Z) \leq \tau) + P(Q(Z) < \tau, \widetilde{Q}^*(Z) \geq \tau)$ can also be constructed but are of little practical value. Since $\{\widetilde{Q}^*(Z) \leq \tau\} = \{\widetilde{Q}(Z) \leq \tau\} \cup \{\widetilde{Q}(Z) > \tau, Q(Z) \leq \tau\}$ and that $\{\widetilde{Q}(Z) > \tau\} = \{\widetilde{Q}(Z) > \tau, Q(Z) > \tau\}$,

$$\begin{aligned} &P(Q(Z) > \tau, \widetilde{Q}^*(Z) \leq \tau) + P(Q(Z) < \tau, \widetilde{Q}^*(Z) \geq \tau) \\ &\leq P(Q(Z) > \tau, \widetilde{Q}(Z) \leq \tau) \leq P(|Q(Z) - \widetilde{Q}(Z)| \geq 0), \end{aligned}$$

however, $P(|Q(Z) - \widetilde{Q}(Z)| \geq 0)$ is usually not available in practice. We now investigate the performance of the multifidelity surrogate model applied to the setup in Example 4.

Algorithm 1 Multifidelity surrogate model $\widetilde{Q}^*(Z)$

Let $Q(Z)$ be the QoI and $\widetilde{Q}(Z)$ be a surrogate model.
For a sample z of Z:
if $\widetilde{Q}(z) > \tau$ **then**
    Evaluate $Q(z)$.
    Set $\widetilde{Q}^*(z) = Q(z)$.
else
    Set $\widetilde{Q}^*(z) = \widetilde{Q}(z)$.
end if

Example 5 *Consider the setup in Example 4 and let $\widetilde{Q}^{adapt}_{max}(Z)$ be the surrogate model approximation to $Q_{max}(Z)$ resulting from the adaptive construction outlined in Section 3.1. Denote by $\widetilde{Q}^*_{max}$ the multifidelity surrogate model according to Algorithm 1. We examine the performance of $\widetilde{Q}^*_{max}(Z)$ in classifying whether $Q_{max}(Z) > \tau$ or $Q_{max}(Z) \leq \tau$.*

By generating Monte Carlo samples of $\widetilde{Q}^*_{max}(Z)$ according to Algorithm 1, we can readily investigate the convergence of the Monte Carlo estimate for the precision, $P(Q_{max}(Z) > \tau \,|\, \widetilde{Q}^{adapt}_{max} > \tau)$, at no additional computational expense. This is displayed in the left panel of Figure 11 which plots this quantity as a function of the number of samples of $\widetilde{Q}^{adapt}_{max}(Z) > \tau$. For completeness, the Monte Carlo estimate of the recall, $P(\widetilde{Q}^{adapt}_{max} > \tau \,|\, Q_{max}(Z) > \tau)$, as a function of the number of samples of $Q_{max}(Z) > \tau$ is plotted in the right panel, although this is acquired through extra evaluations of the full model. Both plots show that the precision and recall estimates are close to convergence after 100 samples of the conditioned event.

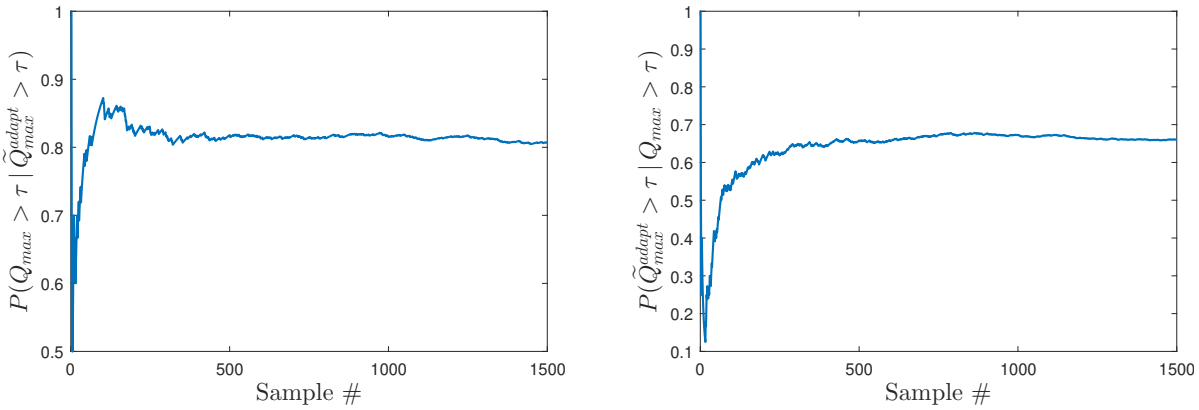

Figure 11: Monte Carlo estimates of precision (left panel) and recall (right panel) as a function of the number of samples of $\widetilde{Q}^{adapt}_{max}(Z) > \tau$ and $Q_{max}(Z) > \tau$, respectively.

As in Example 4, we tabulate the confusion matrix for $\widetilde{Q}^*_{max}(Z)$ as a classifier which we display in Tables 6 and 7 using the same test set of 10000 and 50000 samples of Z. The only difference in these confusion matrices is that the entries in the second column are updated; in particular, the false positives ($\{\widetilde{Q}^*_{max}(Z) > \tau\} \cap \{Q_{max}(Z) \leq \tau\}$) are now 0. The calculated metrics from Table 6 are 1 for precision, 0.6111 for recall, and 0.0007 for the failure rate. For Table 7, they are 1 for precision, 0.6667 for recall, and 0.0008 for the failure rate. Despite the perfect precision and low failure rate, the recall is still low for $\widetilde{Q}^*_{max}(Z)$ to be qualified as a desirable classifier that can aid in predicting if $Q_{max}(Z)$ is large. Even the recall rate were high, certain features of the multifidelity surrogate may be undesirable in certain applications. While $\widetilde{Q}^*_{max}(Z)$ can be utilized to generate samples of $A(x, Z) \,|\, Q_{max}(Z) > \tau$ to study its conditional law, a classification scheme that does not invoke the full model anymore may be preferred for purposes of prediction. We therefore investigate the use of a machine learning classifier in Section 4 to complement the multifidelity surrogate.

$\widetilde{Q}^*_{max}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	11	0	11
Negative	7	9982	9989
Total	18	9982	10000

Table 6: Confusion matrix for $\widetilde{Q}^*_{max}(Z)$ based on 10000 samples of Z.

$\widetilde{Q}^*_{max}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	84	0	84
Negative	42	49874	49916
Total	126	49874	50000

Table 7: Confusion matrix for $\widetilde{Q}^*_{max}(Z)$ based on 50000 samples of Z.

4 Machine learning classifiers

In the previous sections, we investigated various classification schemes to accomplish our objectives. Section 2 underscored that physics-based indicators from $A(x, \omega)$, if they can be found, require numerous full model solves for validation and calibration. In Section 3, we saw that the multifidelity surrogate model still necessitated infrequent full model evaluations for prediction. In this section, we leverage on the strengths of various classification schemes through a two-stage approach that builds on Section 3 and commonly used machine learning classifiers. We first assess in Section 4.1 the performance of only using a machine learning classifer for rare event simulations. Section 4.2 examines the general framework we propose.

4.1 Support vector machines (SVMs) for rare event simulations

In this subsection, we attempt to address our objectives of identifying input random field samples that yield extreme response through machine learning classifiers. While the classification schemes studied previously incorporated knowledge about the system equations at varying levels, classifiers traditionally used in machine learning are constructed solely based on data. The classification process is generally comprised of two components: a model for the classifier and 3 types of data sets, namely, training, validation, and test data.

An example of a model is the SVM classifier given by a linear combination of kernels weighted by unknown coefficients. These coefficients result from solving an optimization problem while the kernels depend on specified parameters, referred to as hyperparameters. Values of the unknown coefficients and of the hyperparameters are then obtained from the training and validation data. More specifically, let the training set be represented by samples $\{(z_i, y_i)\}_{i=1}^M$ of the random vector (Z, Y) with $Z \in \mathbb{R}^d$ and $Y = \pm 1$ where the $Y = +1$ label refers to $Q_{max}(Z) > \tau$ with $Y = -1$ signifying otherwise. The classifier corresponding to the optimal coefficients partitions these samples into two classes according to their label y_i under some criterion.

Since the optimal coefficients rely on specified hyperparameters, changing the values of the latter may alter the former and hence the classifier itself. Validation data therefore serve to tune the hyperparameter values. The performance of the classifier is finally reported using the test data to produce statistics such as the confusion matrices in Section 3. Applied to our setup, the classifier can be used to determine the label of an arbitrary sample of Z and conclude if it resides in the failure region. Unlike surrogate models which approximate the QoI, the output of a machine learning classifier depends on the type used: logistic regression returns the probability that a sample belongs to a class, SVMs evaluate the value of a function whose sign (+ or −) indicates class membership, etc.

Various algorithms exists for classification [18] yet we only concentrate on support vector machines due to its ease of implementation and geometric interpretation; the discussion in Section 4.2 is not catered to a specific classification algorithm. A review of the important concepts of SVMs [19, 20, 18] that we allude to, including its hyperparameters, is presented in Appendix A. We only implement SVMs with Gaussian kernels relying on the scale parameter γ. Despite the relative popularity of SVMs in classification problems, its performance may be hindered in rare event applications due to the scarcity of rare event samples in the training data. This implies multiple solves of the full model to guarantee a sufficient number of training samples in the failure region. To resolve this, existing work [4] suggests variants of latin hypercube sampling (LHS) [21] to obtain training data for the SVM that are adequately spread throughout the bounded range of Z. We

therefore assess the performance of using an SVM trained on latin hypercube samples to classify whether or not a sample of Z corresponds to a rare event. Two synthetic examples are examined below in which the failure region is comprised of 2 disjoint sets to mimic the setting in Example 4. In these examples, latin hypercube sampling can be applied directly because the range of Z is a hypercube, unlike that in Example 4.

Example 6 *Suppose that $Z = (Z_1, \ldots, Z_{10})$ where $Z_i \sim U(0,1)$ are iid random variables. Let $F = \{z \in [0,1]^{10} \,|\, \|z\| \leq 0.94\} \cup \{z \in [0,1]^{10} \,|\, \|z-1\| \leq 0.94\}$ be the failure region such that $Y = 1$ if $Z \in F$ and $Y = -1$, otherwise. For a fixed computational budget of 10000 samples for the training data, we investigate the performance of an SVM classifier trained on latin hypercube samples in predicting if $z \in F$ or $z \notin F$ for any sample z of Z.*

The probability of failure for this example is $P(Z \in F) = \frac{1}{2^9}\frac{\pi^5 (0.94)^1 0}{\Gamma(6)} \approx 0.00268$ calculated using the formula for the volume of a ball in n-dimensions where $\Gamma(\cdot)$ is the gamma function. This is comparable to that of Example 4. Since samples generated through LHS are not fixed, 1000 sets of 10000 latin hypercube samples were produced and we selected the set with the most number of samples contained in F which was 48 samples shared between the 2 connected components of F. The SVM was trained using the Python package `scikit-learn` [22]. We varied the regularization parameter C in (A.2) and the kernel parameter γ using the suggested ranges $1e^{-3} \leq C \leq 1e^{6}$ and $1e^{-6} \leq \gamma \leq 1e^{3}$ with 10 evenly spaced values for each on a logarithmic scale. For each pair of values for C and γ, the trained SVM classifier was applied to 100000 test samples of Z distinct from the training data and simulated according to the true distribution of Z to compute precision and recall metrics. These are summarized in Figure 12. A validation data set was not utilized since we desired to examine metrics across a full spectrum of hyperparameters. The white squares for precision in the figure signify that the value was `nan`. To understand why this occurs, let $f(z)$ be the target classifier such that $f(z) > 0$ whenever $z \in F$ and $f(z) \leq 0$ otherwise and let $f^{SVM}(z)$ be the SVM classifier which predicts that $z \in F$ if $f^{SVM}(z) > 0$. Precision in this case is expressed as $P(f(Z) > 0 \,|\, f^{SVM}(Z) > 0)$ and is undefined whenever $f^{SVM}(z) \leq 0$ almost surely, ie. the trained SVM classifier always predicts that $z \notin F$ which is due to the imbalance between the classes. From Figure 12, a good combination of attained metric values is 0.83 for precision and 0.82 for recall achieved at $\gamma = 1$ and $C = 10$.

This example illustrates the situation when there is a large imbalance in the number of training samples in and outside the failure region. In the next example, we demonstrate that the SVM can still underperform even if this issue does not arise.

Example 7 *Suppose instead that $Z = (Z_1, \ldots, Z_{10})$ where: $Z_i = F^{-1}(\Phi(G_i))$, G_i's are zero-mean, unit-variance Gaussian random variables with $E[G_i G_j] = 0.8$ for $1 \leq i, j \leq 10, i \neq j$, $\Phi(\cdot)$ is the standard normal cdf and $F(\cdot)$ is the cdf of Beta$(10,10)$. The failure region is set to $F = \{z \in [0,1]^{10} \,|\, \|z - (\underbrace{1, \ldots, 1}_{5}, 0, \ldots, 0)\| \leq 0.1.4515\} \cup \{z \in [0,1]^{10} \,|\, \|z - (0, \ldots, 0, \underbrace{1, \ldots, 1}_{5})\| \leq 1.4515\}$ with $Y = 1$ if $Z \in F$ and $Y = -1$, otherwise. As before, the objective is to train an SVM classifier based on 10000 latin hypercube samples to predict if $z \in F$ or $z \notin F$ for any sample z of Z.*

The probability of failure is 0.0026, estimated using 100000 Monte Carlo samples of Z. Only 1 set of 10000 LHS was generated since 1886 of these were contained in F, a substantial increase from that of Example 6. The reason for this is that in high dimensions, there is usually an inverse relationship between the Lebesgue and probability measure: sets of low probability occupy a large volume and vice versa. Figure 13 compiles the precision and recall rates for various combinations of C and γ computed using a test set of 100000 samples of Z simulated according to the true distribution of Z. A validation set was not utilized due to similar reasons stated above. A good balance between the two metrics is 0.66 for precision and 0.63 for recall attained at $C = 1, \gamma = 10$. The SVM classifier in this case suffers from low precision rates. Since most of the samples in the test set are from the high probability region of Z and outside F, they are concentrated in a small volume of the domain wherein the classifier is incorrectly predicting that they belong to F.

Examples 6 and 7 illustrate that SVMs trained on latin hypercube samples alone may not yield sufficiently high rates of precision and recall for rare event applications, unless the number of training samples is increased. This is regardless of the number of training samples present in the failure region. Furthermore,

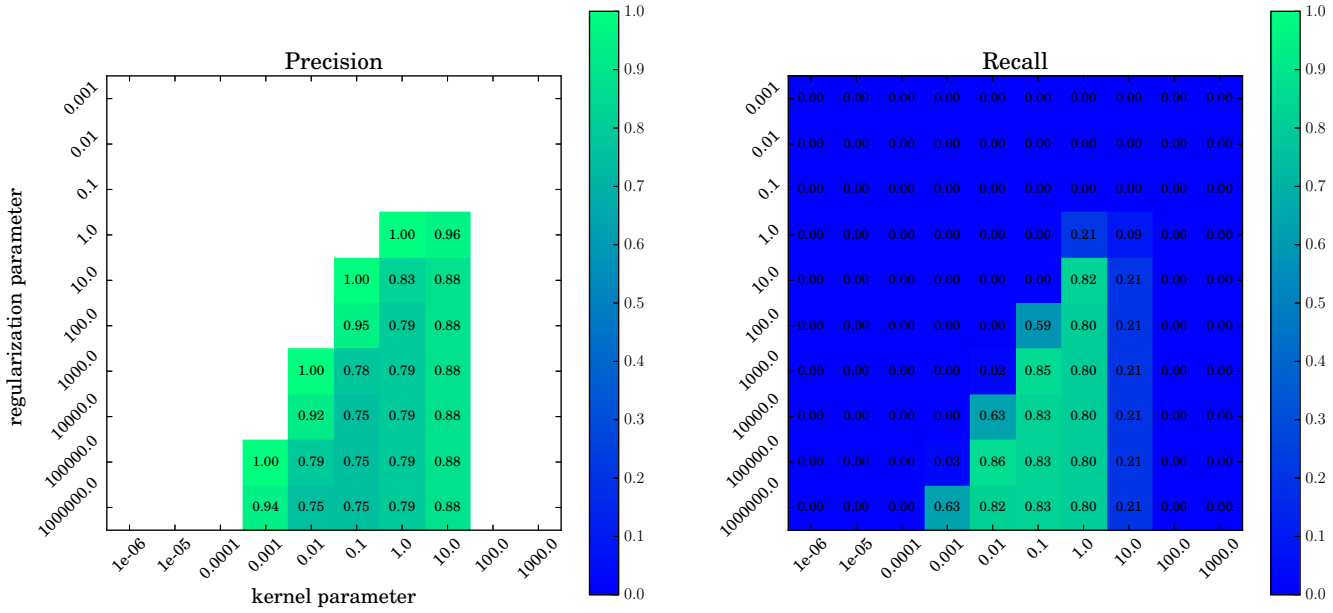

Figure 12: Precision (left) and recall (right) metrics as a function of the regularization C and kernel γ parameters for the SVM classifier in Example 6 based on 100000 test data.

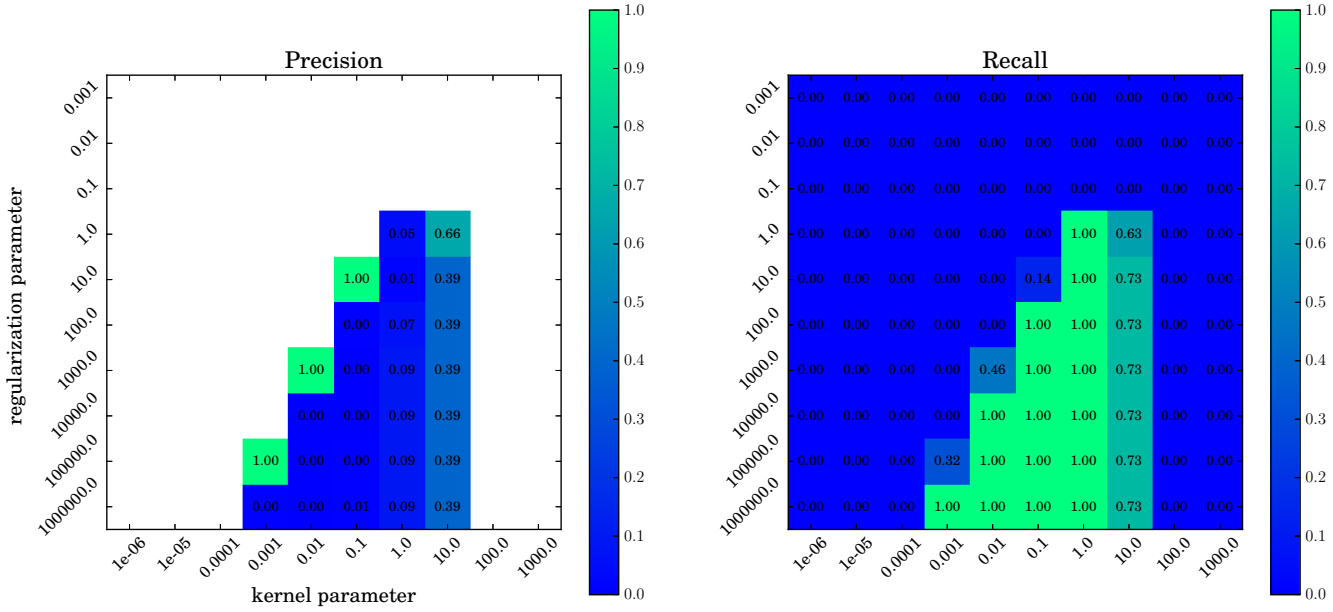

Figure 13: Precision (left) and recall (right) metrics as a function of the regularization C and kernel γ parameters for the SVM classifier in Example 7 based on 100000 test data.

this sampling approach is only directly valid in hypercube domains. We note that it is possible that using other sample points (design of experiments) to train the SVM will produce improved rates for the precision

and recall in the above examples. However, it is not clear which design of experiment is the most appropriate for each problem and counterexamples can always be found for which a particular design underperforms. Our objective is also on the development of a methodology rather than finding which particular tool should be used to implement the methodology. We now proceed to search for an efficient method to obtain data to train machine learning classifiers. For scenarios in which the data arises from computationally expensive forward models, we accomplish this through a multifidelity surrogate model to produce training data at a low computational cost.

4.2 Combining multifidelity surrogates and SVMs

We now introduce the two-stage approach to identify samples of $A(x, Z)$ which yield large QoI. In the first stage, samples of $A(x, Z)$ and their approximate QoI $Q(Z)$ are efficiently generated through a multifidelity surrogate model defined in Algorithm 1. These samples then serve as data to train an SVM that classifies whether $Q > \tau$ in the second stage. This framework enables one to aquire enough samples of $A(x, Z) \,|\, Q(Z) > \tau$ to study its conditional distribution and also offers a scheme for prediction that eliminates the need to evaluate the full model. In view of the examples in Section 4.1, the advantage of simulating the training data in this manner is that a sufficiently large number of samples in the failure region can be collected within a specified computational budget while training samples outside the failure region are available at negligible cost. Algorithm 2 details the proposed procedure for generating training data through a multifidelity surrogate. It is assumed that a surrogate approximation $\widetilde{Q}(Z)$ to the QoI $Q(Z)$ has already been constructed. Following the notation in Section 4.1, $Y = 1$ if $Z \in F$, the failure region, and $Y = -1$ otherwise.

Algorithm 2 Generating training data using a multifidelity surrogate model

Let η_1, η_{-1} be the minimum number of samples needed for (Z, Y) s.t. $Y = 1$ and $Y = -1$, respectively.
Set $D = \{\}$ be the set of samples generated.
Denote by $i_{\eta_1}, i_{\eta_{-1}}$ the current number of samples in D corresponding to $Y = 1$ and $Y = -1$.
while $i_{\eta_1} < \eta_1$ **do**
  Generate a sample z of Z and evaluate $\widetilde{Q}(z)$.
  if $\widetilde{Q}(z) > \tau$ **then**
    Evaluate $Q(z)$.
    if $Q(z) > \tau$ **then**
      $D \leftarrow D \cup \{(z, 1)\}$
      $i_{\eta_1} = i_{\eta_1} + 1$
    else
      $D \leftarrow D \cup \{(z, -1)\}$
      $i_{\eta_{-1}} = i_{\eta_{-1}} + 1$
    end if
  end if
end while
while $i_{\eta_{-1}} < \eta_{-1}$ **do**
  Generate a sample z of Z and evaluate $\widetilde{Q}(z)$.
  if $\widetilde{Q}(z) \leq \tau$ **then**
    $D \leftarrow D \cup \{(z, -1)\}$
    $i_{\eta_{-1}} = i_{\eta_{-1}} + 1$
  end if
end while

Succinctly, Algorithm 2 is just an application of Algorithm 1 in which we prioritize generating samples in the failure region until the target number of failure samples is reached because this requires evaluating the full model. A large number of samples outside the failure region can then be produced as desired due to the negligible cost of evaluating $\widetilde{Q}(Z)$ (lines 18-22 of Algorithm 2). Simulating training samples in this manner implies that there are noisy samples since the labels of some samples are flipped from $+1$ to -1. These false negatives are due to the fact that the full model is not evaluated whenever $\widetilde{Q}(z) \leq \tau$. However, since $P(\widetilde{Q}(Z) \leq \tau, Q(Z) > \tau) \leq P(Q(Z) > \tau)$, we expect the number of samples with flipped labels to be insignificant especially if the failure probability is very low in magnitude.

The work [23] surveys existing classification schemes that deal with flipped labels in the training data. However, it is usually assumed that samples with flipped labels are independent from each other which is not applicable in our setting. An example of such work for SVMs includes [24] which resolves the noise in the training data by solving a non-convex optimization problem. We prefer to adhere to the standard SVM classifier in Appendix A and argue that the regularization parameter is sufficient in handling the small number of samples with flipped labels as in our case. We now apply the two-stage approach to the following example to supplement the performance of the multifidelity surrogate in Section 3.2.

Example 8 *We build on the setup in Examples 4 and 5 in which the multifidelity surrogate according to Algorithm 1 was constructed using the full model $Q_{max}(Z)$ and the surrogate model $\widetilde{Q}_{max}^{adapt}(Z)$ following the adaptive construction for the SROM-based surrogate. Samples of (Z, Y) are then generated according to Algorithm 2 to train an SVM classifier $\widetilde{Q}_{max}^{SVM}(Z)$ with $\widetilde{Q}_{max}^{SVM}(z) = 1$ if $z \in F$ and $\widetilde{Q}_{max}^{SVM}(z) = -1$ otherwise. We examine the performance of $\widetilde{Q}_{max}^{SVM}(Z)$ in predicting samples of Z (and consequently $A(x, Z)$) that yield large $Q_{max}(Z)$.*

Regardless of how the training data are generated, an imbalance in the number of samples present in and out of the failure region will always exist since the failure domain has low probability. To address this, a common strategy is to oversample the minority class or undersample the majority class [25]. In our experiments, we simulated 5021 samples in the failure region for the training data. Following Algorithm 2, this resulted in 6265 evaluations of the full model $Q_{max}(Z)$ of which 1244 corresponded to $Q_{max}(Z) \leq \tau$. In addition to these 1244 samples, 68756 samples outside the failure region were produced at negligible cost using $\widetilde{Q}_{max}^{adapt}$ (lines 18-22 of Algorithm 2) so that a total of 70000 samples reside outside F. Recall that an insignificant proportion of the 70000 samples have flipped labels. We therefore have that the proportion of samples in the training data contained in the failure region is $\frac{5021}{70000+5021} \approx 0.0669$ which is substantially larger than $P(Q_{max}(Z) > \tau) \approx 0.00234$. Notice that our oversampling scheme does not entail producing duplicates of existing samples in the training data pertaining to the minority class.

To tune the hyperparameters C and γ for the SVM, we did not pursue k-fold cross validation since a few samples of the training data have flipped labels and more importantly, the distribution of our training set do not reflect the true distribution of Z. Note that the latter reason is also applicable for SVMs trained on latin hypercube samples. Instead, we generated a validation set comprised of 50000 samples $\{z_k^{valid}, y_k^{valid}\}_{k=1}^{50000}$ of (Z, Y) that requires full model evaluations, is simulated according to the true distribution of (Z, Y), and is not contaminated with noise. Our specific validation set includes 112 samples in F. The hyperparameters chosen are the ones that yield a good combination of precision and recall rates when the trained SVM is applied to the validation data. Figure 14 summarizes these metrics for certain values of C, γ. It is seen that $C = 7500$ and $\gamma = 0.0005$ offer a good combination with 0.94 for precision and 0.88 for recall and are thus the hyperparameter values we select.

To report the performance of the classifier, the trained SVM with the chosen hyperparameters is applied to test data. This consists of the same set of 10000 and 50000 samples of Z (and the corresponding evaluations of the full model $Q_{max}(Z)$) invoked to construct the confusion matrices in Examples 4 and 5. The samples in the test set are distinct from that of the validation set and are also distributed according to the true law of (Z, Y) with no samples possessing flipped labels unlike in the training data.

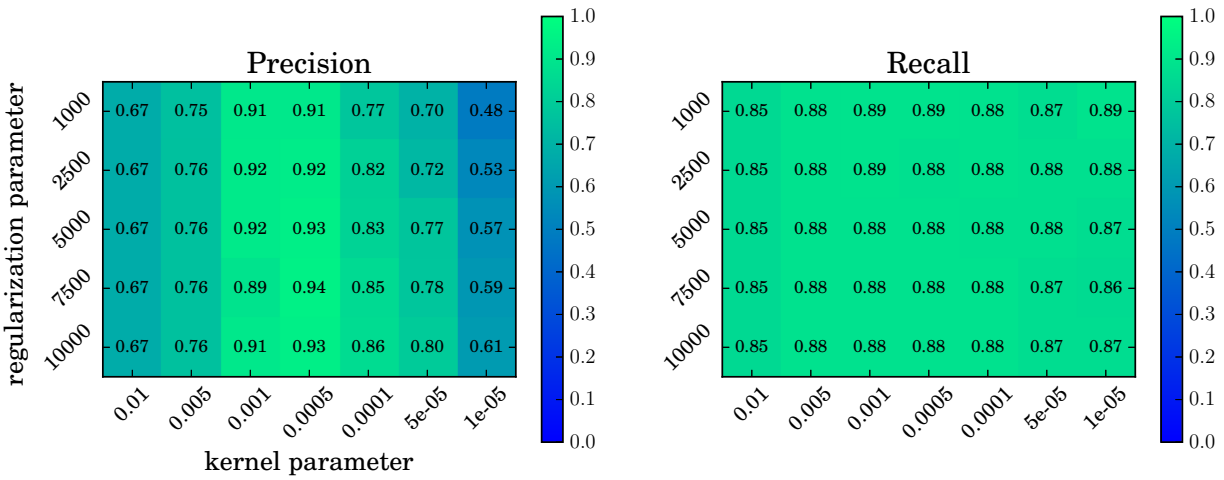

Figure 14: Precision (left) and recall (right) metrics of the trained SVM classifier based on the validation data for certain values of C and γ.

$\widetilde{Q}_{max}^{SVM}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	17	3	20
Negative	1	9979	9980
Total	18	9982	10000

Table 8: Confusion matrix for $\widetilde{Q}_{max}^{SVM}(Z)$ based on 10000 samples of Z.

$\widetilde{Q}_{max}^{SVM}(Z)$ \ $Q_{max}(Z)$	Positive	Negative	Total
Positive	118	13	131
Negative	8	49861	49869
Total	126	49874	50000

Table 9: Confusion matrix for $\widetilde{Q}_{max}^{SVM}(Z)$ based on 50000 samples of Z.

Tables 8 and 9 summarize the confusion matrices of the SVM classifier based on 10000 and 50000 test samples, respectively. From Table 8, we deduce 0.85 for precision, 0.9444 for recall, and 0.1501 for the failure rate while from Table 9, we obtain 0.9008 for precision, 0.9365 for recall, and 0.0994 for the failure rate. Observe that the balance in these metrics is a significant improvement compared to those attained in Examples 4 and 5 and that the failure rate alone as introduced in [7] is insufficient and can be misleading as a metric since it does not incorporate recall. Only a total of 10305 evaluations of the full model to gather the training data were required, 4040 of which are due to constructing the surrogate model $\widetilde{Q}_{max}$ in Section 3.

We now investigate the effect of the samples with flipped labels present in our training data. Figure 15 displays a boxplot of the false negatives, samples whose labels were flipped from $+1$ to -1 due to the multifidelity surrogate in Algorithm 2. The left panel plots their values according to $\widetilde{Q}_{max}^{adapt}(Z)$ while the right panel plots the $Q_{max}(Z)$ values. These only accounted for 61 out of the 70000 training samples outside the failure region. Notice that these false negatives are mostly concentrated close to the boundary of the failure region as the Q_{max} values on the right panel indicate. This underscores that confusion matrices alone do not offer the full perspective on a classifier: it is possible that for a sample z, $\widetilde{Q}_{max}^{adapt}(z) \approx Q_{max}(z)$ yet $\widetilde{Q}_{max}^{adapt}(z) > \tau$ and $Q_{max}(z) \leq \tau$ or vice versa. From Appendix A, it is therefore not surprising that the value of the regularization parameter C is relatively large which enables these false negative samples (-1 label) close to the true boundary of the failure region to be located on the side of the boundary of the SVM classifier pertaining to the $+1$ class. Indeed, let $f(z)$ be the separating manifold in Appendix A whose sign indicates whether $\widetilde{Q}_{max}^{SVM}$ is $+1$ or -1. Evaluating $f(z)$ associated to the trained SVM classifier at these 61 false negative samples resulted in 32 of these with $f(z) > 0$, i.e. $\widetilde{Q}_{max}^{SVM}(z) = 1$.

Finally, we remark that metrics we obtained in Table 8 and 9 are not necessarily the most optimal that could possibly be found. This is because many parameters in Example 8 could be adjusted such as the ratio of the samples in the failure region to the samples outside it in the training data. The former can be increased

at additional computational expense while the latter can be increased or decreased at negligible cost since they are produced using the surrogate model.

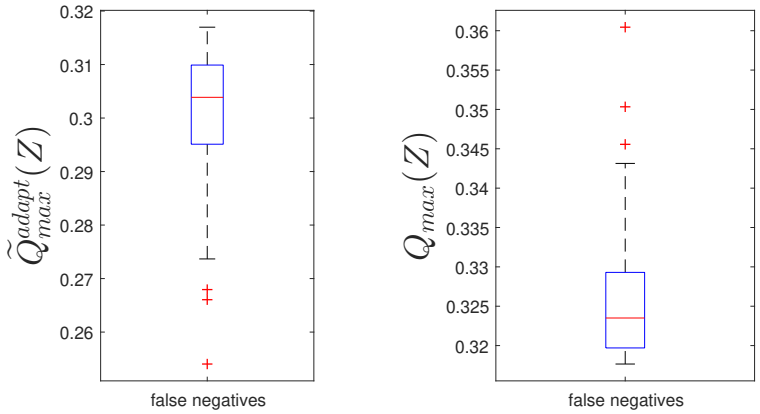

Figure 15: Box plot of the false negatives in the training data in Example 8 for their corresponding $\widetilde{Q}_{max}^{adapt}(Z)$ (left panel) and $Q_{max}(Z)$ (right panel) values.

We conclude this section by underscoring the disparity in the performance of the SVM classifier if the training data were generated using standard Monte Carlo simulation as Example 9 demonstrates.

Example 9 *We revisit Example 8 except that instead of generating training data according to the proposed scheme in Algorithm 2, they are produced simply through Monte Carlo simulation based on the true law of Z. The effect of the scarcity of training samples in the failure region on the SVM classifier is examined.*

The training data for the SVM was gathered from 50000 samples of Z and $Q_{max}(Z)$ of which 115 resided in the failure region. Only 1 set of Monte Carlo samples was generated for the purposes of illustration. The hyperparameter values were varied in the ranges $1e^{-3} \leq C \leq 1e^{6}$ and $1e^{-6} \leq \gamma \leq 1e^{3}$ with 10 evenly spaced values for each on a logarithmic scale. For each combination of C and γ values, the SVM was applied to the same test set of 50000 samples of Z utilized in Examples 4, 5, and 8. We did not make use of a validation set in order to inspect the metrics for a spectrum of hyperparameter values. The precision and recall rates are presented in Figure 16 with the white squares indicating `nan` values. A good balance between the two rates is attained at $C = 100, \gamma = 0.001$ with 0.86 for precision and 0.71 for recall. The fact that the classifier achieves high rates of precision but low rates of recall for various combinations of hyperparameter values signifies that it is unable to capture a large scope of the failure region: it is likely that the high probability regions of the failure domain identified by the SVM is mostly contained in the true failure domain but that the high probability region of the true failure domain is largely absent in the failure domain identified by the SVM.

The above examples serves to highlight the synergy that can be obtained by combining a multifidelity surrogate to generate training data efficiently and a machine learning classifier to identify samples of $A(x, \omega)$ that yield large QoI. Implicit in them is the availability of validation data that have no flipped labels and that are distributed according to its true distribution to tune the SVM hyperparameters. Obtaining such data efficiently will be considered in future work.

5 Discussion

In this section, we make some remarks on our proposed framework and mention future directions.

By invoking surrogate models and machine learning classifiers, Sections 3 and 4 sought an approach that is universal and can be applied to various problem setups unlike the physics-based indicators in Section 2. This requires that the input random field $A(x, Z)$ is parametric, depending on a random vector Z, although

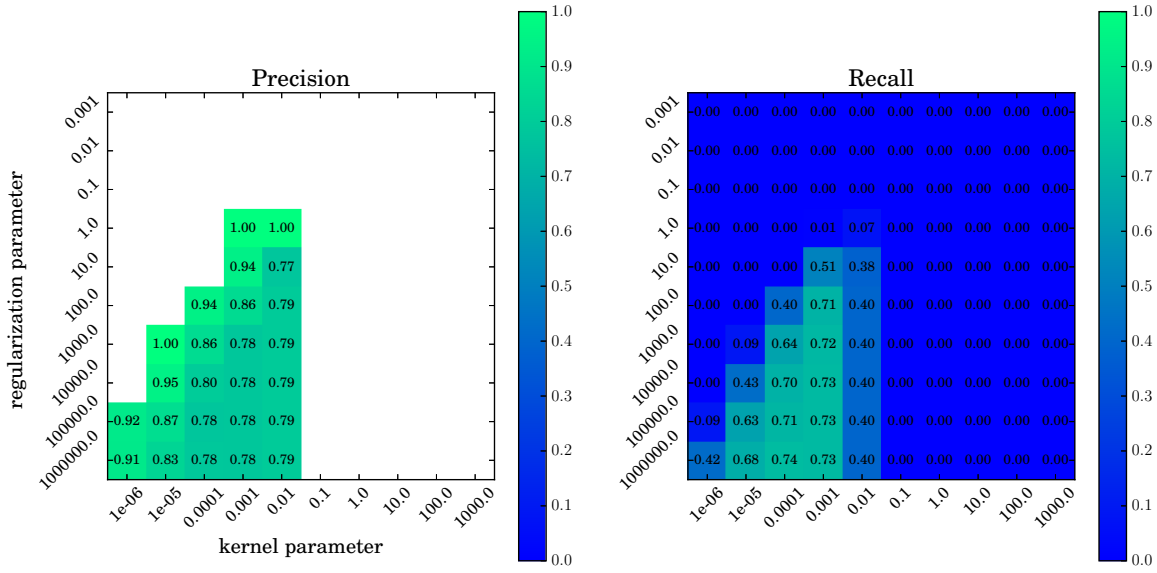

Figure 16: Precision (left) and recall (right) as a function of the regularization C and kernel γ parameters for the SVM classifier in Example 9.

$A(x, Z)$ need not be linear with respect to Z as in our implementation in (3.2). In more general situations, the input random field $A(x, \omega)$ is infinite-dimensional which is then approximated by a parametric field $A(x, Z(\omega))$. Denote by $Q(\omega), Q(Z(\omega))$ the respective quantities of interest resulting from these 2 random fields and let $\widetilde{Q}(Z(\omega))$ be the surrogate approximation to $Q(Z(\omega))$. Our proposed framework is restricted to the setting where the given input random field is parametric with the goal of minimizing the discrepancy between the sets $\{Z \,|\, Q(Z) > \tau\}$ and $\{Z \,|\, \widetilde{Q}(Z) > \tau\}$. A future direction of this work would address the more general setting involving the input random field $A(x, \omega)$ with the goal of making the sets $\{\omega \,|\, Q(\omega) > \tau\}$ and $\{\omega \,|\, \widetilde{Q}(Z(\omega)) > \tau\}$ as close as possible.

In our proposed framework, which consists of a surrogate model and an SVM classifier, there is flexibility in the choice of the surrogate. In fact, Algorithm 2 holds regardless of the surrogate used. Our objective is not to identify which surrogate is the most appropriate for each problem. We only require the surrogate to converge to the true model in $L^\infty(\Omega)$ or in $L^p(\Omega)$ for large p as discussed in Section 3.1. This is satisfied, among the various types of surrogates available, by sparse grid stochastic collocation [16] and by Gaussian process regression [26] under some conditions. The chosen surrogate must also be implemented in such a way that the approximation is not only accurate in regions of the parameter space with high probability since the input samples causing rare events are in the low probability region.

Finally, a few remarks are in order regarding the computational cost of the proposed framework in Section 4.2. As mentioned previously, the generation of SVM training data using the multifidelity surrogate and generation of validation data required 10305 and 50000 full model evaluations, respectively, while the generation of test data required a maximum of 50000 evaluations. In comparison, surrogate models that can accurately estimate failure probabilities in reliability engineering do not incur as much computational expense. This only underscores the unique challenges of the problem we consider: our objective can be considered as an inverse problem while the probability of failure is a global quantity which requires fewer full model evaluations. Observe that the bulk of the cost above stems from generation of validation and test data, not from the generation of training data which involves the surrogate construction. Validation and test data are standard components of statistics and machine learning workflow [18]. In this work, a large

number of validation and test samples are needed in order to have sufficient samples in the failure region because these occur with small probability. To reduce the need for a validation set, k-fold cross-validation is a technique employed in statistics and machine learning in which several subsets of the training data are repurposed as validation data. This is not suitable for the approach we pursued since the training data generated do not adhere to the true distribution of Z as explained in Section 4.2. For k-fold cross validation to be applicable here, the training set must be simulated using Monte Carlo samples of Z as performed in Example 9 but as we noticed, this yielded low rates of precision and recall. Designing a framework to reduce the cost of generating the validation and test set is the subject of future work.

6 Conclusion

In this work, we studied the problem of identifying samples of the input random field which yield large quantities of interest. This presents a shift from the traditional objectives in reliability engineering which is mostly concerned with computing probabilities of failure. While such quantities are useful in applications, they serve as global measures in that they do not offer clues as to what types of inputs cause failure nor how to design inputs that do not lead to such events.

Several classification schemes were investigated to achieve our objectives. These included physics-based indicators which are quantities computed from the input random field that signal the occurrence of extreme events. However, in practice, indicators are problem-dependent and may be difficult to derive unless a strong physical intuition about the system is available. In addition, a large number of samples of the rare event and hence full model evaluations is needed to verify and calibrate the indicator. This motivated the search of an alternative classification scheme that can be generalized to various contexts. We therefore examined a multifidelity approximation to the quantity of interest which is comprised of the surrogate model and infrequent evaluations of the full model. Functioning as a classifier that determines if an input random field sample leads to large QoI, the multifidelity surrogate in our specific example achieved high rates of precision yet low rates of recall. Since recurring to infrequent full model evaluations may still be undesirable, we proposed a two-stage approach to identify the samples of interest where in the first step, training data are efficiently gathered using the multifidelity approximation and a machine learning classifier is trained on these samples in the second step. We demonstrated the synergy of these 2 methods in obtaining high rates of precision and recall compared to other options for producing training data such as Monte Carlo simulation and latin hypercube sampling.

Acknowledgements

We are grateful to the anonymous reviewers and the editors for their suggestions and clarifications which have improved the quality of the manuscript.

References

[1] J. Morio, M. Balesdent, D. Jacquemart, and C. Vergé, "A survey of rare event simulation methods for static input–output models," *Simulation Modelling Practice and Theory*, vol. 49, pp. 287–304, 2014.

[2] J. Li and D. Xiu, "Evaluation of failure probability via surrogate models," *Journal of Computational Physics*, vol. 229, no. 23, pp. 8966–8980, 2010.

[3] T. Butler and T. Wildey, "Utilizing adjoint-based error estimates for surrogate models to accurately predict probabilities of events," *International Journal for Uncertainty Quantification*, vol. 8, no. 2, pp. 143–159, 2018.

[4] A. Basudhar, S. Missoum, and A. H. Sanchez, "Limit state function identification using support vector machines for discontinuous responses and disjoint failure domains," *Probabilistic Engineering Mechanics*, vol. 23, no. 1, pp. 1–11, 2008.

[5] B. L. Boyce, B. C. Salzbrenner, J. M. Rodelas, L. P. Swiler, J. D. Madison, B. H. Jared, and Y.-L. Shen, "Extreme-value statistics reveal rare failure-critical defects in additive manufacturing," *Advanced Engineering Materials*, vol. 19, no. 8, p. 1700102, 2017.

[6] M. Farazmand and T. Sapsis, "A variational approach to probing extreme events in turbulent dynamical systems," *Science Advances*, vol. 3, no. 9, p. e1701533, 2017.

[7] M. Farazmand and T. Sapsis, "Extreme events: Mechanisms and prediction," 2018. arXiv:1803.06277v1.

[8] B. Echard, N. Gayton, and M. Lemaire, "AK-MCS: An active learning reliability method combining kriging and monte carlo simulation," *Structural Safety*, vol. 33, pp. 145–154, Mar. 2011.

[9] M. Grigoriu, *Stochastic calculus. Applications in science and engineering.* Boston: Birkhäuser, 2002.

[10] M. Grigoriu, *Stochastic systems. Uncertainty quantification and propagation.* London: Springer Ser. Reliab. Eng., Springer, 2012.

[11] M. Grigoriu, "Parametric models for samples of random functions," *Journal of Computational Physics*, vol. 297, pp. 47–71, 2015.

[12] M. Grigoriu, "Material responses at micro- and macro-scales," *Computational Materials Science*, vol. 107, pp. 190–203, 2015.

[13] R. V. Field and M. Grigoriu, "Convergence properties of polynomial chaos approximations for l^2-random variables," *Sandia Report SAND2007-1262*, 2007.

[14] R. Field and M. Grigoriu, "On the accuracy of the polynomial chaos approximation," *J. Comput. Phys.*, vol. 209, pp. 617–642, 2005.

[15] M. Grigoriu, "Response statistics for random heterogeneous microstructures," *SIAM/ASA J. Uncertainty Quantification*, vol. 2, pp. 252–275, 2014.

[16] W. I. T. Uy and M. D. Grigoriu, "An adaptive method for solving stochastic equations based on interpolants over voronoi cells," *Probabilistic Engineering Mechanics*, vol. 51, pp. 23–41, 2018.

[17] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[18] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning.* Springer New York, 2009.

[19] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods.* Cambridge University Press, 2000.

[20] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning (Adaptive Computation and Machine Learning series).* The MIT Press, 2018.

[21] M. Stein, "Large sample properties of simulations using latin hypercube sampling," *Technometrics*, vol. 29, no. 2, pp. 143–151, 1987.

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[23] B. Frenay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 845–869, 2014.

[24] G. Stempfel and L. Ralaivola, "Learning SVMs from sloppily labeled data," in *Artificial Neural Networks – ICANN 2009*, pp. 884–893, Springer Berlin Heidelberg, 2009.

[25] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, "An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics," *Information Sciences*, vol. 250, pp. 113–141, Nov. 2013.

[26] H. Wendland, *Scattered Data Approximation*. Cambridge University Press, Dec. 2004.

A Review of support vector machines

In this section, the important concepts of support vector machines (SVM) as referred to in Section 4 are reviewed. Additional information such as theoretical aspects about errors can be consulted in [19, 20, 18]. Let $\{(z_i, y_i)\}_{i=1}^M$ be samples of the random vector (Z, Y) which is distributed according to $(Z, Y) \sim \mathcal{D}$ with $Z \in \mathbb{R}^d$ and $Y = \pm 1$. SVMs attempt to discover a function $f(z)$ that separates $\{z_i\}_{i=1}^M$ into 2 classes according to the value of y_i. Once found, it can be used to predict to which class an arbitrary sample z of Z belongs.

Suppose that the data is linearly separable, i.e. $\{z_i\}_{i=1}^M$ can be partitioned into two disjoint sets by a $(d-1)$-dimensional hyperplane $f(z) = \beta_0 + \beta^T z$ where $\beta_0 \in \mathbb{R}$, $\beta, z \in \mathbb{R}^d$. If the constants β_0, β are known, SVMs classify data by evaluating $f(z)$ and assigning z to the positive class ($y = 1$) if $f(z) > 0$ or to the negative class ($y = -1$) otherwise. These constants are chosen to maximize the Euclidean distance between the hyperplane $f(z)$ and the sample z_i closest to it. This requirement can be recast into a convex optimization problem given by

$$\begin{aligned} \underset{\beta_0,\beta}{\text{minimize}} \quad & \frac{1}{2}\|\beta\|^2 \\ \text{subject to} \quad & y_i(\beta^T z_i + \beta_0) \geq 1,\ i = 1, \ldots, M. \end{aligned} \tag{A.1}$$

where the constraints signify that 1) the separating hyperplane $f(z)$ correctly segregates the data into 2 classes and that 2) each sample z_i is at least a distance of $\frac{1}{\|\beta\|}$ from the resulting $f(z)$. However, if the data is not linearly separable, i.e. no such hyperplane exists, (A.1) can be modified by introducing slack variables $\{\epsilon_i\}_{i=1}^M$ so that $f(z)$ only correctly classifies most of the samples. The modified optimization problem now reads as

$$\begin{aligned} \underset{\beta_0,\beta}{\text{minimize}} \quad & \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^M \epsilon_i \\ \text{subject to} \quad & y_i(\beta^T z_i + \beta_0) \geq 1 - \epsilon_i,\ i = 1, \ldots, M, \\ & \epsilon_i \geq 0,\ i = 1, \ldots, M, \end{aligned} \tag{A.2}$$

where C is a regularization parameter that must be specified. Depending on the value of ϵ_i, the updated constraints permit the hyperplane to misclassify some samples ($\epsilon_i > 1$) or allow samples to be very near the hyperplane, i.e. a distance less than $\frac{1}{\|\beta\|}$, despite being classified correctly ($\epsilon_i \leq 1$). The parameter C controls the number of such samples: in general, large C means a smaller penalty term $\sum_{i=1}^M \epsilon_i$ or smaller values for ϵ_i, $i = 1, \ldots, M$ so that misclassified samples in $\{(z_i, y_i)\}_{i=1}^M$ mostly occur near the separating hyperplane $f(z)$ and vice versa [18, p. 418, Figure 12.1].

In practice, it may be more convenient to solve the dual problem of (A.2) which lends additional geometric interpretation. The dual problem is expressed as

$$\begin{aligned} \underset{\alpha_1,\ldots,\alpha_M \in \mathbb{R}}{\text{maximize}} \quad & \sum_{i=1}^M \alpha_i - \frac{1}{2}\sum_{i=1}^M \sum_{j=1}^M y_i y_j \alpha_i \alpha_j z_i^T z_j \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C,\ i = 1, \ldots, M, \\ & \sum_{i=1}^M \alpha_i y_i = 0. \end{aligned} \tag{A.3}$$

From the Karush-Kuhn-Tucker conditions, it can be concluded that $\beta = \sum_{i=1}^{M} \alpha_i y_i z_i$ so that

$$f(z) = \beta_0 + \beta^T z = \beta_0 + \sum_{i=1}^{M} \alpha_i y_i z_i^T z. \tag{A.4}$$

Furthermore, if $\alpha_i = 0$, $y_i f(z_i) \geq 1$, whereas if $\alpha_i = C$, $y_i f(z_i) \leq 1$, while $0 < \alpha_i < C$ implies that $y_i f(z_i) = 1$. These conditions demonstrate that $f(z)$ only depends on the samples z_i for which $\alpha_i \neq 0$ or whose distance from $f(z)$ is at most $\frac{1}{\|\beta\|}$, i.e. the support vectors, and that it is not altered by the remaining samples in the data.

In more complex classification problems, the data may often be separated more adequately by a nonlinear manifold. SVMs handle this by mapping the input z into a feature space $\psi(z) = (\psi_1(z), \ldots, \psi_D(z)) \in \mathbb{R}^D$ in which the data is now linearly separable. It turns out that the mapping $\psi(z)$ need not be known explicitly; it suffices to define a kernel $K(x, z)$ where $\psi(z)$ must satisfy $\psi(x)^T \psi(z) = K(x, z)$. The above discussion on separating data using hyperplanes still holds except that (A.4) becomes $f(z) = \beta_0 + \sum_{i=1}^{M} \alpha_i y_i K(z_i, z)$ while the objective function in (A.3) is rewritten into $\sum_{i=1}^{M} \alpha_i - \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{M} y_i y_j \alpha_i \alpha_j K(z_i, z_j)$. A commonly used kernel is the Gaussian kernel $K(x, z) = \exp\left(-\frac{\|x-z\|^2}{2\gamma^2}\right)$ for specified $\gamma > 0$.

In summary, constructing an SVM entails solving a convex optimization problem to obtain the parameters β, β_0 assuming that the hyperparameter values such as C and the kernel parameter γ are fixed. A standard procedure is to split the data into 3 disjoint sets: the training data from which β, β_0 are estimated for fixed hyperparameter values, the validation data for tuning the hyperparameters, and the test data for reporting the performance of the SVM beyond the training data. An alternative is to combine the training and validation data and employ cross-validation to identify the parameters and hyperparameters.