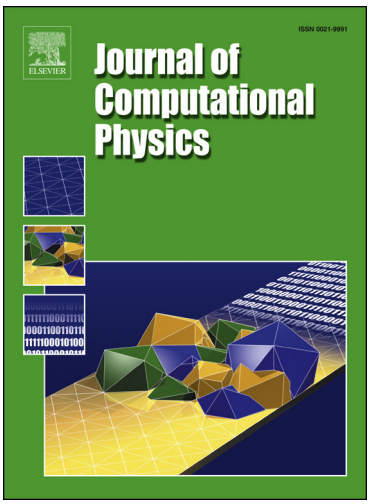
Journal Pre-proof



Nonlinear sparse Bayesian learning for physics-based models

Rimple Sandhu, Mohammad Khalil, Chris Pettit, Dominique Poirel, Abhijit Sarkar

PII: S0021-9991(20)30502-7
DOI: https://doi.org/10.1016/j.jcp.2020.109728
Reference: YJCPH 109728

To appear in: *Journal of Computational Physics*

Received date: 16 December 2019
Revised date: 5 July 2020
Accepted date: 17 July 2020

Please cite this article as: R. Sandhu et al., Nonlinear sparse Bayesian learning for physics-based models, *J. Comput. Phys.* (2020), 109728, doi: https://doi.org/10.1016/j.jcp.2020.109728.

Highlights

- A semi-analytical nonlinear sparse Bayesian algorithm is developed.
- This algorithm permits the inclusion of prior parameter knowledge.
- Numerical examples demonstrate the usefullness of the proposed algorithm.

Nonlinear sparse Bayesian learning for physics-based models

Rimple Sandhu[a,**], Mohammad Khalil[b,*], Chris Pettit[c], Dominique Poirel[d], Abhijit Sarkar[a,]

[a] *Department of Civil & Environmental Engineering, Carleton University, Ottawa, ON, Canada*
[b] *Quantitative Modeling & Analysis Department, Sandia National Laboratories, Livermore, CA, United States*
[c] *Department of Aerospace Engineering, United States Naval Academy, Annapolis, MD, United States*
[d] *Department of Mechanical & Aerospace Engineering, Royal Military College of Canada, Kingston, ON, Canada*

Abstract

This paper addresses the issue of overfitting while calibrating unknown parameters of over-parameterized physics-based models with noisy and incomplete observations. A semi-analytical Bayesian framework of nonlinear sparse Bayesian learning (NSBL) is proposed to identify sparsity among model parameters during Bayesian inversion. NSBL offers significant advantages over machine learning algorithm of sparse Bayesian learning (SBL) for physics-based models, such as 1) the likelihood function or the posterior parameter distribution is not required to be Gaussian, and 2) prior parameter knowledge is incorporated into sparse learning (i.e. not all parameters are treated as questionable). NSBL employs the concept of automatic relevance determination (ARD) to facilitate sparsity among questionable parameters through parameterized prior distributions. The analytical tractability of NSBL is enabled by employing Gaussian ARD priors and by building a Gaussian mixture-model approximation of the posterior parameter distribution that excludes the contribution of ARD priors. Subsequently, type-II maximum likelihood is executed using Newton's method whereby the evidence and its gradient and Hessian information are computed in a semi-analytical fashion. We show numerically and analytically that SBL is a special case of NSBL for linear regression models. Subsequently, a linear regression example involving multimodality in both parameter posterior pdf and model evidence is considered to demonstrate the performance of NSBL in cases where SBL is inapplicable. Next, NSBL is applied to identify sparsity among the damping coefficients of a mass-spring-damper model of a shear building frame. These numerical studies demonstrate the robustness and efficiency of NSBL in alleviating overfitting during Bayesian inversion of nonlinear physics-based models.

Keywords: Inverse problems, sparse learning, Bayesian inference, automatic relevance determination, Gaussian mixture-model, Bayesian model selection, physics-based modelling

1. Introduction

Bayesian inference has gained widespread acceptance in solving the ill-conditioned inverse problem of assimilating noisy observations with imperfect mathematical models, resulting in a posterior probability density function (pdf) of unknown model parameters [1–10]. Incomplete knowledge of underlying physics and the increasing demand to enhance model predictive capabilities often result in over-parameterized models (more unknown parameters than required), which suffer from *overfitting* during Bayesian inversion. This

*Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

**Currently at National Renewable Energy Laboratory, USA
Corresponding author. Tel.: +1 613520 2600x6320; fax: +1 613 520 3951
Email address: abhijit.sarkar@carleton.ca (Abhijit Sarkar)

paper is devoted to resolving this issue of overfitting during Bayesian inversion of nonlinear-in-parameter models (observations nonlinearly related to unknown parameters). For dynamical systems, the unknown time-invariant parameters are considered as model parameters, while the unknown time-varying parameters are considered encapsulated in the state vector. We also assume that significant prior knowledge exists regarding some unknown parameters, a scenario prevalent among models derived from laws-of-physics. We hereby refer to such models as physics-based models.

Previously, the authors have attempted to resolve the issue of overfitting for a nonlinear aeroelastic system by executing an evidence-based Bayesian model comparison on a set of equations nested under an over-parameterized, nonlinear, stochastic differential equation [11]. However, Bayesian model comparison among nested models (or equations) was found to be sensitive to 1) the width of prior parameter pdf, and 2) the choice of nested models considered for comparison. It has been well established that Bayesian model comparison tends to favor simpler models with increasing prior widths [9, 12]. Recently, the authors [13] exploited the concept of automatic relevance determination (ARD) to alleviate these practical issues by converting the Bayesian model comparison task into a sparse learning problem, which then transformed the model comparison problem from a discrete model domain into a continuous hyperparameter (ARD-prior parameters) domain. This transformation enabled the implicit comparison of all models nested under an over-parameterized model. To handle non-Gaussian posterior parameter pdfs resulting from nonlinear models, the model evidence was estimated using Markov chain Monte Carlo (MCMC) posterior parameter samples, followed by a gradient-free maximization of the evidence estimate to obtain the sparse model parameter structure. This approach is in contrast to data-driven techniques of sparse Bayesian learning (SBL) [14, 15] and Bayesian compressive sensing (BCS) [16] which are only applicable to linear-in-parameter models and conjugate priors.

Despite its many practical advantages, the MCMC-powered ARD approach [13] involved a sampling-within-optimization step, which significantly degraded the overall computational efficiency of the inversion process. This degradation intensified with an increasing model dimension (state or parameter space) and with an increasing cost of likelihood function computation. For instance, when dealing with nonlinear dynamical systems and sparse observations, the likelihood computation often requires a subtask of sampling-based state estimation (or data-assimilation), which further exacerbates the computational efficiency of the sampling-within-optimization task [13, 17]. Besides, the accuracy of sparse learning relies on the goodness of MCMC sampling. Since the posterior parameter pdf varies with a changing hyperparameter of ARD prior, the subsequent tuning and monitoring of the MCMC sampler during evidence maximization is rendered impractical.

In this paper, we propose a semi-analytical Bayesian inversion framework that aims to address these computational issues with the MCMC-powered ARD approach [13] while retaining its practical benefits. We call this new framework *nonlinear sparse Bayesian learning* (NSBL) since it is partially motivated by the analytical Bayesian apparatus of SBL, but applies to nonlinear physics-based models (unlike SBL). The key difference between NSBL and our previous ARD approach [13] is that the Bayesian entities (evidence, posterior parameter pdf) are available semi-analytically (in terms of Gaussian kernels). This semi-analytical tractability of the Bayesian apparatus is powered by a Gaussian mixture-model (GMM) approximation of the entity consisting of the product of likelihood function and the known prior pdf of *a priori* relevant parameters (i.e parameters with significant prior knowledge). Consequently, the sampling-within-optimization step of the previous ARD-based approach [13] is replaced with Newton's iteration that exploits the semi-analytically tractable gradient and Hessian information to expedite evidence maximization. Figure 1 provides an overview of the NSBL algorithm.

NSBL is in contrast to the previous Bayesian inversion approaches reported in the literature involving cheap surrogates for the computationally intensive model or the likelihood function or the posterior pdf [18–21]. Most notably, Marzouk *et al.* [18] exploited an intrusive stochastic spectral technique to reformulate the governing equations using a prior-informed polynomial chaos expansion (PCE) of model parameters and then sampling the PCE coefficients instead of model parameters. Also, Galbally *et al.* [21] proposed the projection of a high-fidelity model on a reduced subspace using proper orthogonal decomposition to expedite MCMC sampling in high-dimensional parameter space. The underlying goal of these approaches has been to compute the posterior parameter pdf faster. In contrast, NSBL is primarily aimed at identifying

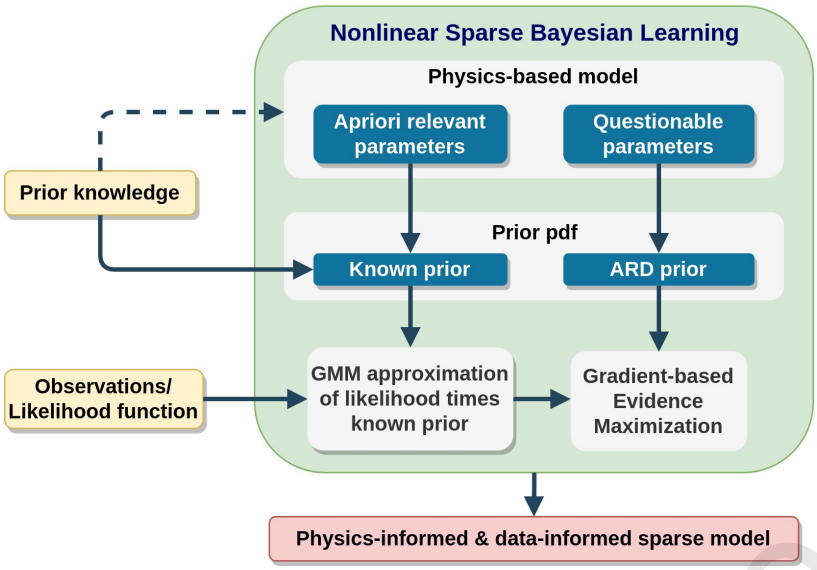


Figure 1: Overview of the NSBL algorithm.

sparse parameter representation among physics-based models. The fact that the posterior parameter pdf is available analytically through NSBL is an added benefit. Also, we do not offer NSBL as a replacement for SBL or BCS for linear-in-parameter models. NSBL is primarily aimed towards physics-based applications where SBL and BCS are inapplicable.

To this end, the following contributions are reported in this paper (listed section-wise):

- In Section 2 we provide a detailed mathematical derivation into the semi-analytical apparatus of NSBL, followed by the numerical implementation details regarding GMM construction and Newton's method for optimizing evidence. We also show analytically that SBL is a special case of NSBL for linear-in-parameter models while Gaussian ARD priors are assigned to all unknown parameters.
- In Section 3.1, we consider a linear regression setting involving the construction of a sparse polynomial chaos expansion (PCE) surrogate for the strongly nonlinear Ishigami function. This numerical exercise is aimed at validating the semi-analytical formulation of NSBL against SBL since both the algorithms are expected to produce similar results in a linear regression setting. Also, computationally efficiency of NSBL is contrasted with SBL and BCS when dealing with high-dimensional models.
- In Section 3.2, we consider a nontrivial linear regression setting wherein the posterior parameter pdf and the model evidence are both multimodal. These special circumstances are synthetically generated using a multimodal prior pdf on a regression model parameter. This numerical investigation highlights the applicability of NSBL to physics-based inverse problems involving multimodality in both parameter and hyperparameter space. SBL and BCS are inapplicable to such cases.
- In Section 3.3, we consider a structural dynamics example consisting of a three degree-of-freedom (dof) mass-spring-damper model of a shear building frame. NSBL is applied to identify the sparse damping structure of the three-dof system using a sparse, noisy and incomplete realization of floor displacement during free vibration. This numerical exercise demonstrates the benefits of NSBL in alleviating overfitting during the inversion of nonlinear-in-parameter differential equations.

2. Methodology: Nonlinear sparse Bayesian learning

Consider that a system model $f : \boldsymbol{\phi} \mapsto \mathbf{y}$ is proposed to model a physical system, where the model operator f maps the unknown model parameter vector $\boldsymbol{\phi} \in \mathbb{R}^{N_\phi}$ to the observed entity $\mathbf{y} \in \mathbb{R}^{N_y}$. The N_d

number of observations of $\mathbf{y}$ are denoted as $\mathcal{D}$. The observations $\mathcal{D}$ are allowed to be noisy, sparse, and incomplete (i.e. the entire system state is not measured). It is assumed that the likelihood function $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$ is known for any $\boldsymbol{\phi}$ value, and the estimation of hidden variables (or unobserved state variables) is encapsulated within the likelihood function. Hence, for the sake of inversion, the likelihood function is considered known for a given $\boldsymbol{\phi}$ value, while the model parameter vector $\boldsymbol{\phi}$ is considered the unknown. The goal of NSBL is to eliminate redundant model parameters and obtain a sparse representation of $\boldsymbol{\phi}$ during Bayesian inversion, leading to an optimally-fitted predictive model. The reader is referred to Section 3.3 wherein Figure 11 and Figure 14 contrasts the predictive performance of an overfitted model with an optimally-fitted model, respectively. Next, we detail the Bayesian setup that empowers the semi-analytical apparatus of NSBL.

2.1. Hybrid prior pdf

Following our previous work [13], the unknown parameter vector is decomposed as $\boldsymbol{\phi} = \{\boldsymbol{\phi}_\alpha, \boldsymbol{\phi}_{\text{-}\alpha}\}$ (in that order), where $\boldsymbol{\phi}_\alpha \in \mathbb{R}^{N_\alpha}$ contains parameters with no prior knowledge (and hence questionable), and $\boldsymbol{\phi}_{\text{-}\alpha} \in \mathbb{R}^{N_\phi - N_\alpha}$ contains parameters with a known prior pdf $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. Following SBL, $\boldsymbol{\phi}_\alpha$ is assigned a Gaussian ARD prior $\mathrm{p}(\boldsymbol{\phi}_\alpha|\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})$, where $\mathbf{A} = \mathrm{Diag}(\boldsymbol{\alpha})$ is the prior precision matrix, and $\boldsymbol{\alpha} \in \mathbb{R}^{N_\alpha}$ is the unknown hyperparameter vector. Notice that the matrix $\mathbf{A}$ is diagonal, implying the prior independence among questionable parameters. The marginal ARD prior pdf for parameter $\phi_i \in \boldsymbol{\phi}_\alpha$ can be written as $\mathrm{p}(\phi_i|\alpha_i) = \mathcal{N}(\phi_i|0, \alpha_i^{-1})$. Note that hyperparameter α_i controls the contribution of parameter ϕ_i in the model since setting $\alpha_i = \infty$ would force both the prior and posterior pdf of ϕ_i to be a Dirac-delta pdf $\mathcal{N}(\phi_i|0, 0)$ centered at zero. In other words, $\boldsymbol{\alpha}$ controls the complexity of the model.

The choice of Gaussian ARD priors is dictated by two reasons. First, Gaussian priors enable the semi-analytical computation of Bayesian entities. Second, using the principle of maximum entropy, a Gaussian pdf contains minimum information or maximum entropy for a random variable with finite mean and finite variance [10]. This property of a Gaussian ARD prior will ensure minimum interference from the modeler on the posterior distribution of sparse relevant parameters identified through sparse learning. This property is desired since a zero-mean prior pdf tends to pull the posterior parameter space towards the origin. Employing a Gaussian ARD prior over other types of priors will minimize this pull.

The joint prior pdf of $\boldsymbol{\phi}$ is summarized as

$$\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha}) = \mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})\mathrm{p}(\boldsymbol{\phi}_\alpha|\boldsymbol{\alpha}) = \mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1}). \tag{1}$$

This hybrid prior pdf enables sparse learning of questionable parameters in $\boldsymbol{\phi}_\alpha$ through ARD prior $\mathrm{p}(\boldsymbol{\phi}_\alpha|\boldsymbol{\alpha})$ while incorporating prior knowledge about $\boldsymbol{\phi}_{\text{-}\alpha}$ through $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$.

2.2. Gaussian mixture-model approximation

Given observations $\mathcal{D}$ and hyperparameter vector $\boldsymbol{\alpha}$, the parameter posterior pdf $\mathrm{p}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha})$ is obtained using Bayesian inference as

$$\mathrm{p}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha}) = \frac{\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha})}{\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})} = \frac{\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})}{\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})}, \tag{2}$$

where $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$ is the likelihood function, $\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})$ is the model evidence (or marginal likelihood or type-II likelihood), and $\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha})$ is the joint prior pdf from Eq. (1). NSBL operates by building a GMM approximation for the entity $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ in Eq. (2) as

$$\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha}) \approx \sum_{k=1}^{K} a^{(k)} \mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}), \tag{3}$$

where K is the total number of kernels, $a^{(k)} \in \mathbb{R}$ is the kernel coefficient ($a^{(k)} > 0$) and $\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ is a Gaussian pdf with mean vector $\boldsymbol{\mu}^{(k)} \in \mathbb{R}^{N_\phi}$ and covariance matrix $\boldsymbol{\Sigma}^{(k)} \in \mathbb{R}^{N_\phi \times N_\phi}$.

Note that only one kernel ($K = 1$) is sufficient in Eq. (3) under the special circumstances of 1) linear regression with Gaussian likelihood $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$ and Gaussian prior $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ (same setup as SBL), or 2) 'large'

number of observations in $\mathcal{D}$ such that $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ can be approximated as a Gaussian (Laplace approximation [22]). For any other case, the approximation in Eq. (3) will require more than one Gaussian kernel and unknown entities $a^{(k)}$, $\boldsymbol{\mu}^{(k)}$ and $\boldsymbol{\Sigma}^{(k)}$ need to estimated numerically. For instance, a kernel density estimation (KDE) approximation using Gaussian kernels can be employed to construct the GMM in Eq. (3). Section 2.6 provides specific implementation details behind Eq. (3).

The GMM approximation in Eq. (3) offer the following benefits for sparse learning among physics-based models:

- The use of Gaussian kernels in Eq. (3), in combination with Gaussian ARD priors, enables analytical evaluation of parameter posterior pdf $\mathrm{p}(\boldsymbol{\phi}|\mathcal{D},\boldsymbol{\alpha})$, model evidence $\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})$, and the gradient and Hessian of evidence with respect to hyperparameters $\boldsymbol{\alpha}$. The semi-analytical apparatus of NSBL is powered by this analytical tractability of Bayesian entities.

- Entity $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ in Eq. (3) is much well-behaved (in terms of identifiability [4]) than the likelihood function $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$ due to the regularization effect of the known prior $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. In other words, the known prior $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ helps restrict $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ in the $\boldsymbol{\phi}$ space that makes physical sense and complies with the prior knowledge. This property is desired when generating stationary samples from $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ for the sake of constructing a GMM. This is one of the key differences between NSBL and purely data-based techniques that solely rely on the likelihood function.

- Since $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ is independent of $\boldsymbol{\alpha}$, the GMM approximation in Eq. (3) is only needed to be built once for the sake of sparse learning.

- The kernel-based GMM approximation in Eq. (3) allows for representing non-Gaussian or multimodal $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ encountered in engineering applications. This property of the GMM allows for the handling of multimodal or skewed likelihood function $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$, or a non-Gaussian prior pdf $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. Although Eq. (3) can handle any $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ choice, it is considered best practice to classify parameters with limited prior knowledge as 'questionable', instead of assigning them a uniform prior with broad support. We expect that the GMM approximation will be best useful and easier built when $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ regularizes $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$.

- Eq. (3) facilitates direct handling of the likelihood function instead of the model $f(\boldsymbol{\phi})$. This model-free property of NSBL is desired for inverse problems where the model $f(\boldsymbol{\phi})$ is only available as a black-box numerical solver. In other words, NSBL algorithm only needs access to the likelihood function, and a closed-form expression for the physical model is not required.

Next, we define the sparse learning problem in mathematical terms.

2.3. Sparse learning optimization problem

Using the hierarchical Bayes approach, the hyperparameter posterior pdf $\mathrm{p}(\boldsymbol{\alpha}|\mathcal{D})$ is obtained as [10]

$$\mathrm{p}(\boldsymbol{\alpha}|\mathcal{D}) = \frac{\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})\mathrm{p}(\boldsymbol{\alpha})}{\mathrm{p}(\mathcal{D})} \propto \mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})\mathrm{p}(\boldsymbol{\alpha}), \tag{4}$$

where $\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})$ is the model evidence from Eq. (2), $\mathrm{p}(\boldsymbol{\alpha})$ is the hyper-prior (prior for hyperparameter $\boldsymbol{\alpha}$), and $\mathrm{p}(\mathcal{D})$ is just a normalization constant for a fixed $\mathcal{D}$. Note that our interest in the sparsity of $\boldsymbol{\phi}_\alpha$ requires us to obtain an optimal $\boldsymbol{\alpha}$ value and not the entire posterior distribution $\mathrm{p}(\boldsymbol{\alpha}|\mathcal{D})$ from Eq. (4). The *maximum a posteriori* (MAP) estimate $\boldsymbol{\alpha}^{\text{map}}$ provides such an optimal choice, obtained by maximizing the hyperparameter posterior $\mathrm{p}(\boldsymbol{\alpha}|\mathcal{D})$ from Eq. (4) as

$$\boldsymbol{\alpha}^{\text{map}} = \arg\max_{\boldsymbol{\alpha}}\{\mathrm{p}(\boldsymbol{\alpha}|\mathcal{D})\} = \arg\max_{\boldsymbol{\alpha}}\{\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})\mathrm{p}(\boldsymbol{\alpha})\}. \tag{5}$$

For a sparse $\boldsymbol{\phi}_\alpha$, many α_i^{map} will approach infinity, thereby forcing the marginal posterior pdf of ϕ_i to be a Dirac-delta function centered at zero. A finite α_i^{map} will imply a relevant parameter $\phi_i \in \boldsymbol{\phi}_\alpha$.

Following SBL [14], the hyperparameters in $\boldsymbol{\alpha}$ are assumed to be *a priori* independent and the marginal hyperprior pdf $\mathrm{p}(\alpha_i)$ is chosen to be a Gamma distribution. The joint hyperprior $\mathrm{p}(\boldsymbol{\alpha})$ is written as

$$\mathrm{p}(\boldsymbol{\alpha}) = \prod_{i=1}^{N_\alpha} \mathrm{p}(\alpha_i) = \prod_{i=1}^{N_\alpha} \mathcal{G}(\alpha_i|r_i, s_i) = \prod_{i=1}^{N_\alpha} \frac{s_i^{r_i}}{\Gamma(r_i)} \alpha_i^{r_i-1} e^{-s_i\alpha_i}, \tag{6}$$

where $\mathcal{G}(\alpha_i|r_i, s_i)$ denotes a univariate Gamma distribution with shape parameter $r_i > 0$ and rate parameter $s_i > 0$. Aside from enforcing positivity constraint for the precision parameter α_i, a Gamma distribution can be reduced to many simplified informative or non-informative distributions by varying r and s values, as detailed in Table 1. In this work, we will employ Jeffrey's prior (flat prior over $\log \alpha_i$) for all $\alpha_i \in \boldsymbol{\alpha}$, which is obtained by using values of $s \approx 0$ and $r \approx 0$ in Eq. (6). Note that when using a non-informative prior for $\boldsymbol{\alpha}$ (such as Jeffrey's prior), the optimal value of $\boldsymbol{\alpha}^{\mathrm{map}}$ is solely dictated by model evidence $\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})$. This approach of maximizing model evidence is also known as type-II maximum likelihood [10].

Prior type	Parameters in Eq. (6)	Type of pdf	$\mathrm{p}(\alpha)$
Informative	$r = 1,\ s = \lambda$	Exponential	$\lambda e^{-\lambda\alpha}$
	$r > 0,\ s \to \infty$	Dirac-delta	$\delta(\alpha - r/s)$
Non-informative	$r \to 0^+,\ s \to 0^+$	Jeffery's prior	$\mathrm{p}(\alpha) \propto 1/\alpha$ or $\mathrm{p}(\log\alpha) \propto 1$
	$r \to 1^+,\ s \to 0^+$	Flat prior	$\mathrm{p}(\alpha) \propto 1$ or $\mathrm{p}(\log\alpha) \propto \lvert\alpha\rvert$

Table 1: Special cases of a Gamma hyperprior pdf.

Note that Eq. (13) is a non-convex optimization problem for any given combination of likelihood function $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$, prior pdf $\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha})$ and hyperprior pdf $\mathrm{p}(\boldsymbol{\alpha})$ [22]. In the SBL/RVM setup for linear regression models with Gaussian errors, Faul and Tipping [23] showed analytically that $\log \mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})$ has a unique global optimum with respect to an individual hyperparameter α_i (not the entire $\boldsymbol{\alpha}$ vector) when all other hyperparameters are held fixed. SBL and BCS exploited this property to propose a semi-analytical re-estimation procedure derived by setting the gradient of log-evidence with respect to α_i to zero. Faul and Tipping [23] also showed that the optimal $\boldsymbol{\alpha}$ obtained through this re-estimation procedure will be a joint optimum for all α_i. However, the uniqueness of this optimal $\boldsymbol{\alpha}$ was put in question by Faul and Tipping [23], pointing towards the non-convex nature of the evidence optimization. Nevertheless, the models encountered in engineering mechanics are far from linear regression, where SBL and BCS are inapplicable and a unique global optimum of log-evidence with respect to α_i or $\boldsymbol{\alpha}$ is not guaranteed. We will also demonstrate this non-uniqueness of $\boldsymbol{\alpha}^{\mathrm{map}}$ (computed using Eq. (5)) through a numerical example in Section 3.1. Next, we provide a detailed mathematical exposition into the analytical calculation of Bayesian entities.

2.4. Semi-analytical calculation of Bayesian entities

2.4.1. Model evidence

Given the the joint prior pdf in Eq. (1), the model evidence in Eq. (2) is written as

$$\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha}) = \int \mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha}) d\boldsymbol{\phi} = \int \mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})\mathrm{p}(\boldsymbol{\phi}_\alpha|\boldsymbol{\alpha}) d\boldsymbol{\phi}. \tag{7}$$

An estimate $\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha})$ of model evidence is obtained by substituting Eq. (3) and ARD prior $\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})$ in Eq. (7) as

$$\begin{aligned}
\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha}) &= \int \left\{ \sum_{k=1}^{K} a^{(k)} \mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}) \right\} \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1}) d\boldsymbol{\phi} \\
&= \sum_{k=1}^{K} a^{(k)} \int \mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}) \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1}) d\boldsymbol{\phi}.
\end{aligned} \tag{8}$$

Note that the integral in Eq. (8) involves the product of two multivariate Gaussian pdfs with different dimensions ($\boldsymbol{\phi}_\alpha$ vs $\boldsymbol{\phi}$). Using the parameter decomposition $\boldsymbol{\phi} = \{\boldsymbol{\phi}_\alpha, \boldsymbol{\phi}_{\text{-}\alpha}\}$, Eq. (8) is simplified to obtain (details in Appendix B.1)

$$\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha}) = \sum_{k=1}^{K} a^{(k)} \mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)}), \tag{9}$$

where $\mathbf{B}_\alpha^{(k)} = \boldsymbol{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1}$ and $a^{(k)}$, $\boldsymbol{\mu}_\alpha^{(k)}$, and $\boldsymbol{\Sigma}_\alpha^{(k)}$ are obtained from the GMM approximation in Eq. (3) (details in Appendix B.1). As noted from Eq. (9), the dependence of model evidence on $\boldsymbol{\alpha}$ is through matrix $\mathbf{B}_\alpha^{(k)} = \boldsymbol{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1}$ as $\mathbf{A} = \mathrm{Diag}(\boldsymbol{\alpha})$.

This computation of model evidence for varying $\boldsymbol{\alpha}$ without accessing the likelihood function or the model is a powerful tool in itself. This tool offers a significant computational relief for high-dimensional models (large N_ϕ or large N_α) as it eliminates the need for performing time-consuming sampling of posterior parameter pdf (like MCMC) for estimating evidence for varying $\boldsymbol{\alpha}$ values.

2.4.2. Posterior parameter pdf

An estimate $\hat{\mathrm{p}}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha})$ of the posterior parameter pdf is obtained by substituting $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ from Eq. (3) in Eq. (2) to obtain

$$\hat{\mathrm{p}}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha}) = \sum_{k=1}^{K} \frac{a^{(k)} \mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}) \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})}{\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha})}. \tag{10}$$

Substituting $\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ from Eq. (B.2a) reduces Eq. (10) to

$$\begin{aligned}
\hat{\mathrm{p}}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha}) &= \sum_{k=1}^{K} \frac{a^{(k)} \mathcal{N}(\boldsymbol{\phi}_{\text{-}\alpha}|\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}, \tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}) \mathcal{N}(\boldsymbol{\phi}_\alpha|\boldsymbol{\mu}_\alpha^{(k)}, \boldsymbol{\Sigma}_\alpha^{(k)}) \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})}{\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha})} \\
&= \sum_{k=1}^{K} \underbrace{\left(\frac{a^{(k)} \mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)})}{\sum_{r=1}^{K} a^{(r)} \mathcal{N}(\boldsymbol{\mu}_\alpha^{(r)}|\mathbf{0}, \mathbf{B}_\alpha^{(r)})} \right)}_{w^{(k)}} \underbrace{\frac{\mathcal{N}(\boldsymbol{\phi}_{\text{-}\alpha}|\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}, \tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}) \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{m}_\alpha^{(k)}, \mathbf{P}_\alpha^{(k)})}{\mathcal{N}(\boldsymbol{\phi}|\mathbf{m}^{(k)}, \mathbf{P}^{(k)})}}_{},
\end{aligned} \tag{11}$$

where $0 \leq w^{(k)} \leq 1$ is the weight coefficient, $\mathbf{m}^{(k)}$ is the posterior mean of $\boldsymbol{\phi}$, and $\mathbf{P}^{(k)}$ is the posterior covariance of $\boldsymbol{\phi}$; all pertaining to the k^{th} kernel. Notice that the sum of all weight coefficients is one, i.e. $\sum_k w^{(k)} = 1$.

The posterior pdf from Eq. (11) is rewritten in an expanded form as

$$\hat{\mathrm{p}}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha}) = \sum_{k=1}^{K} w^{(k)} \mathcal{N}(\boldsymbol{\phi}|\mathbf{m}^{(k)}, \mathbf{P}^{(k)}) = \sum_{k=1}^{K} w^{(k)} \mathcal{N}\left(\left\{ \begin{matrix} \boldsymbol{\phi}_\alpha \\ \boldsymbol{\phi}_{\text{-}\alpha} \end{matrix} \right\} \middle| \left\{ \begin{matrix} \mathbf{m}_\alpha^{(k)} \\ \mathbf{m}_{\text{-}\alpha}^{(k)} \end{matrix} \right\}, \begin{bmatrix} \mathbf{P}_\alpha^{(k)} & \mathbf{D}^{(k)} \\ (\mathbf{D}^{(k)})^T & \mathbf{P}_{\text{-}\alpha}^{(k)} \end{bmatrix} \right) \tag{12}$$

where $\mathbf{m}_\alpha^{(k)}$ and $\mathbf{P}_\alpha^{(k)}$ are the posterior entities pertaining to $\boldsymbol{\phi}_\alpha$; $\mathbf{m}_{\text{-}\alpha}^{(k)}$ and $\mathbf{P}_{\text{-}\alpha}^{(k)}$ pertain to $\boldsymbol{\phi}_{\text{-}\alpha}$; and $\mathbf{D}^{(k)}$ is the posterior cross-covariance of $\boldsymbol{\phi}_\alpha$ and $\boldsymbol{\phi}_{\text{-}\alpha}$. The semi-analytical calculation of these posterior entities is detailed in Appendix B.2.

2.4.3. Gradient vector and Hessian matrix

To facilitate the semi-analytical calculation of gradient and Hessian, the optimization problem in Eq. (5) is reposed in terms of $\log \mathrm{p}(\boldsymbol{\alpha}|\mathcal{D})$. In addition, the optimization is performed with respect to $\log \boldsymbol{\alpha}$ instead of $\boldsymbol{\alpha}$, which automatically enforces the positivity constraint of $\boldsymbol{\alpha}$ during the optimization.

Following these modifications, the optimization problem in Eq. (5) is rewritten as

$$\begin{aligned}
\log \boldsymbol{\alpha}^{\text{map}} &= \arg\max_{\log \boldsymbol{\alpha}} \{\mathcal{L}(\log \boldsymbol{\alpha})\} = \arg\max_{\log \boldsymbol{\alpha}} \{\log \mathrm{p}(\log \boldsymbol{\alpha}|\mathcal{D})\} \\
&= \arg\max_{\log \boldsymbol{\alpha}} \{\log \hat{\mathrm{p}}(\mathcal{D}|\log \boldsymbol{\alpha}) + \log \mathrm{p}(\log \boldsymbol{\alpha})\},
\end{aligned} \tag{13}$$

where the model evidence estimate $\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})$ is available from Eq. (9), and $\mathcal{L}(\log\boldsymbol{\alpha})$ is the objective function (or log-evidence as a function of $\log\boldsymbol{\alpha}$) that needs to be optimized for identifying the sparse structure of $\boldsymbol{\phi}$.

Given $\mathrm{p}(\boldsymbol{\alpha})$ in Eq. (6), hyperprior $\mathrm{p}(\log\boldsymbol{\alpha})$ required in Eq. (13) is obtained using the univariate transformation of random variables as [24]

$$\mathrm{p}(\log\boldsymbol{\alpha}) = \prod_{i=1}^{N_\alpha} \mathrm{p}(\log\alpha_i) = \prod_{i=1}^{N_\alpha} \frac{s_i^{r_i}}{\Gamma(r_i)} \alpha_i^{r_i} e^{-s_i\alpha_i}. \tag{14}$$

Subsequently, the objective function $\mathcal{L}(\log\boldsymbol{\alpha})$ from Eq. (13) is rewritten using Eq. (14) as (ignoring constant terms)

$$\mathcal{L}(\log\boldsymbol{\alpha}) = \log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha}) + \sum_{i=1}^{N_\alpha} \left(r_i \log\alpha_i - s_i\alpha_i\right). \tag{15}$$

Notice that when using Jeffrey's prior ($\mathrm{p}(\log\alpha_i) \propto 1$) for α_i, hyperprior parameters r_i and s_i will be close to zero, and the objective function in Eq. (15) will be solely dictated by the log-evidence estimator $\log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})$.

The i^{th} element of the gradient vector $\mathbf{J}(\log\boldsymbol{\alpha})$, denoted as $J_i(\log\boldsymbol{\alpha})$, is obtained by differentiating Eq. (15) with respect to $\log\alpha_i$ as

$$\begin{aligned} J_i(\log\boldsymbol{\alpha}) = \frac{\partial\mathcal{L}(\log\boldsymbol{\alpha})}{\partial\log\alpha_i} &= \frac{\partial}{\partial\log\alpha_i}\left\{\log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha}) + \sum_{i=1}^{N_\alpha}\left(r_i\log\alpha_i - s_i\alpha_i\right)\right\} \\ &= \frac{\partial\log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}{\partial\log\alpha_i} + r_i - s_i\alpha_i\,. \end{aligned} \tag{16}$$

The log-evidence $\log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})$ from Eq. (9) is differentiated with respect to $\log\alpha_i$ to obtain

$$\begin{aligned} \frac{\partial\log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}{\partial\log\alpha_i} &= \frac{1}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\frac{\partial}{\partial\log\alpha_i}\left\{\sum_{k=1}^{K} a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})\right\} \\ &= \frac{1}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\sum_{k=1}^{K} a^{(k)}\frac{\partial}{\partial\log\alpha_i}\left\{\exp\left(\log\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})\right)\right\} \\ &= \sum_{k=1}^{K}\underbrace{\left(\frac{a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\right)}_{w^{(k)}}\underbrace{\left(\frac{\partial\log\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\partial\log\alpha_i}\right)}_{v_i^{(k)}}, \end{aligned} \tag{17}$$

where the weight coefficient $w^{(k)}$ has been previously defined in Eq. (11), and factor $v_i^{(k)}$ is analytically tractable and is obtained as (details in Appendix B.3)

$$v_i^{(k)} = -\frac{1}{2}\left\{-1 + \alpha_i P_{ii}^{(k)} + \alpha_i(m_i^{(k)})^2\right\} = \frac{\gamma_i^{(k)} - \alpha_i(m_i^{(k)})^2}{2}\,, \tag{18}$$

where $\gamma_i^{(k)} = 1 - \alpha_i P_{ii}^{(k)}$ is defined as the relevance indicator for parameter $\boldsymbol{\phi}_i$ corresponding to the k^{th} kernel. As demonstrated later in Section 2.5, the relevance indicator $\gamma_i^{(k)}$ provides a quantitative measure for determining the relevancy of ϕ_i for a given value of α_i. Nevertheless, the semi-analytical solution to gradient $J_i(\log\boldsymbol{\alpha})$ is obtained as

$$J_i(\log\boldsymbol{\alpha}) = \left\{\sum_{k=1}^{K}\left(\frac{a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\right)\left(\frac{\gamma_i^{(k)} - \alpha_i(m_i^{(k)})^2}{2}\right)\right\} + r_i - s_i\alpha_i \tag{19a}$$

$$= \sum_{k=1}^{K} w^{(k)} v_i^{(k)} + r_i - s_i\alpha_i, \tag{19b}$$

Similarly, the (i,j) element of the Hessian matrix $\mathbf{H}(\log\boldsymbol{\alpha})$, denoted as $H_{ij}(\log\boldsymbol{\alpha})$, is evaluated by differentiating Eq. (19b) as

$$H_{ij}(\log\boldsymbol{\alpha}) = \frac{\partial^2\mathcal{L}(\log\boldsymbol{\alpha})}{\partial\log\alpha_i\partial\log\alpha_j} = \frac{\partial J_j(\log\boldsymbol{\alpha})}{\partial\log\alpha_i} = \frac{\partial}{\partial\log\alpha_i}\left\{\sum_{k=1}^{K} w^{(k)}v_j^{(k)} + r_j - s_j\alpha_j\right\}$$
$$= \sum_{k=1}^{K}\left\{w^{(k)}\frac{\partial v_j^{(k)}}{\partial\log\alpha_i} + v_j^{(k)}\frac{\partial w^{(k)}}{\partial\log\alpha_i}\right\} - \delta_{ij}s_i\alpha_i \tag{20}$$

where $\delta_{ij}=1$ when $i=j$ and $\delta_{ij}=0$ when $i\neq j$. This final solution to $H_{ij}(\log\boldsymbol{\alpha})$ is obtained as (details in Appendix B.4)

$$H_{ij}(\log\boldsymbol{\alpha}) = \sum_{k=1}^{K}\Bigg[w^{(k)}\left\{\alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)}m_j^{(k)}P_{ij}^{(k)}\right) + \delta_{ij}\left(v_i^{(k)} - \frac{1}{2}\right)\right\}$$
$$+ v_j^{(k)}\left\{w^{(k)}\left(v_i^{(k)} - \bar{v}_i\right)\right\}\Bigg] - \delta_{ij}s_i\alpha_i. \tag{21}$$

Table 2 summarizes the semi-analytical Bayesian framework of NSBL.

Entity	Solution
Parameter decomposition	$\boldsymbol{\phi} = \{\boldsymbol{\phi}_\alpha, \boldsymbol{\phi}_{\text{-}\alpha}\}$
GMM approximation	$\mathrm{p}(\mathcal{D}\|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha}) \approx \sum_{k=1}^{K} a^{(k)}\mathcal{N}(\boldsymbol{\phi}\|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$
Hybrid prior pdf	$\mathrm{p}(\boldsymbol{\phi}\|\boldsymbol{\alpha}) = \mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})\mathrm{p}(\boldsymbol{\phi}_\alpha\|\boldsymbol{\alpha}) = \mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})\mathcal{N}(\boldsymbol{\phi}_\alpha\|\mathbf{0}, \mathbf{A}^{-1})$
Hyperprior	$\mathrm{p}(\boldsymbol{\alpha}) = \prod_{i=1}^{N_\alpha}\mathcal{G}(\alpha_i\|r_i, s_i)$
Model evidence	$\hat{\mathrm{p}}(\mathcal{D}\|\boldsymbol{\alpha}) = \sum_{k=1}^{K} a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}\|\mathbf{0}, \mathbf{B}_\alpha^{(k)})$; $\mathbf{B}_\alpha^{(k)} = \boldsymbol{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1}$
Parameter posterior pdf	$\hat{\mathrm{p}}(\boldsymbol{\phi}\|\mathcal{D}, \boldsymbol{\alpha}) = \sum_{k=1}^{K} w^{(k)}\mathcal{N}(\boldsymbol{\phi}\|\mathbf{m}^{(k)}, \mathbf{P}^{(k)})$; $w^{(k)} = \frac{a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}\|\mathbf{0}, \mathbf{B}_\alpha^{(k)})}{\hat{\mathrm{p}}(\mathcal{D}\|\log\boldsymbol{\alpha})}$
	$\mathbf{m}^{(k)}$ and $\mathbf{P}^{(k)}$ solution in eqs. (12), (B.11), (B.14), (B.4c), (B.4d) and (B.8b)
Objective function	$\mathcal{L}(\log\boldsymbol{\alpha}) = \log\hat{\mathrm{p}}(\mathcal{D}\|\log\boldsymbol{\alpha}) + \sum_{i=1}^{N_\alpha}(r_i\log\alpha_i - s_i\alpha_i)$
Gradient of $\mathcal{L}(\log\boldsymbol{\alpha})$	$J_i(\log\boldsymbol{\alpha}) = \sum_{k=1}^{K} w^{(k)}v_i^{(k)} + r_i - s_i\alpha_i = \bar{v}_i + r_i - s_i\alpha_i$
	$v_i^{(k)} = (\gamma_i^{(k)} - \alpha_i(m_i^{(k)})^2)/2$; $\gamma_i^{(k)} = 1 - \alpha_i P_{ii}^{(k)}$
Hessian of $\mathcal{L}(\log\boldsymbol{\alpha})$	$H_{ij}(\log\boldsymbol{\alpha}) = \sum_{k=1}^{K} w^{(k)}\left\{\alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)}m_j^{(k)}P_{ij}^{(k)}\right) + v_i^{(k)}v_j^{(k)} - \bar{v}_i\bar{v}_j\right\}$
	$+\delta_{ij}\left\{\bar{v}_i - \frac{1}{2} - s_i\alpha_i\right\}$

Table 2: Summary of the analytical Bayesian apparatus of NSBL.

2.5. Relevance indicator

Once the hyperparameter MAP estimate $\log\boldsymbol{\alpha}^{\text{map}}$ is computed by solving the optimization problem in Eq. (13), the relevance of each questionable parameter $\phi_i \in \boldsymbol{\phi}_\alpha$ needs to be determined using the corresponding $\log\alpha_i^{\text{map}}$ values. In SBL [14], the relevance of each questionable parameter ϕ_i was determined using the relevance indicator $\gamma_i = 1 - \alpha_i P_{ii}$ where P_{ii} is the posterior variance of ϕ_i. However, in NSBL,

there exist a relevance indicator $\gamma_i^{(k)}$ pertaining to each Gaussian kernel $a^{(k)}\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ from Eq. (3). The true nature of $\gamma_i^{(k)}$ is revealed by substituting $P_{ii}^{(k)}$ from Eq. (B.4c) to write

$$\gamma_i^{(k)} = 1 - \frac{\alpha_i}{(P_{ii}^{(k)})^{-1}} = 1 - \frac{\alpha_i}{\alpha_i + \{(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}\}_{(i,i)}} \in [0, 1], \tag{22}$$

where $\{(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}\}_{(i,i)}$ is the (i, i) element of precision matrix $(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}$. For an irrelevant parameter ϕ_i, the prior precision α_i will be large (low prior variance) and the posterior will be dictated by the prior so that $(P_{ii}^{(k)})^{-1} \approx \alpha_i$ or $\gamma_i^{(k)} \approx 0$. Alternatively, for a relevant parameter ϕ_i, the posterior precision $(P_{ii}^{(k)})^{-1}$ will be dictated by observations, thereby forcing $P_{ii}^{-1} \approx (\Sigma_i^{(k)})^{-1}$ or $\gamma_i^{(k)} \approx 1$. In other words, $\gamma_i^{(k)}$ indicates the percentage of posterior precision that is due to observations $\mathcal{D}$ in the k^{th} kernel. Also, $\gamma_i^{(k)}$ varies between zero and one and therefore provides a consistent quantitative measure of relevance for $\phi_i \in \boldsymbol{\phi}_\alpha$ according to the k^{th} kernel.

We extend this idea of relevance indicator to a multi-kernel setting by taking root-mean-square (RMS) of K relevance indicators $\gamma_i^{(k)}$ as

$$\gamma_i^{\text{rms}} = \left(\frac{1}{K} \sum_{k=1}^{K} (\gamma_i^{(k)})^2 \right)^{1/2} = \left(\frac{1}{K} \sum_{k=1}^{K} \left(1 - \alpha_i P_{ii}^{(k)} \right)^2 \right)^{1/2}. \tag{23}$$

This summarized measure γ_i^{rms} in Eq. (23) will also vary in range $[0, 1]$ since each $\gamma_i^{(k)} \in [0, 1]$ from Eq. (22). A γ_i^{rms} value close to zero will imply irrelevance and a γ_i^{rms} value close to one will imply relevance. Notice that γ_i^{rms} is a scale-invariant measure since it involves the ratio of prior (α_i) and posterior $((P_{ii}^{(k)})^{-1})$ precision. As a result, γ_i^{rms} should be preferred over α_i values for monitoring parameter relevance during and after optimization. We also propose a predefined tolerance γ^{tol} for γ_i^{rms} where a γ_i^{rms} value greater than γ^{tol} will imply relevance. The impact of γ^{tol} on sparsity levels produced by the NSBL algorithm is investigated in Section 3.1.

2.6. Numerical implementation details

NSBL involves following two numerical tasks: 1) GMM construction of $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ in Eq. (3), and 2) optimization of $\mathcal{L}(\log \boldsymbol{\alpha})$ in Eq. (13). Lets first focus on the task of building a GMM for $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. The unknown entities $a^{(k)}, \boldsymbol{\mu}_\alpha^{(k)}, \boldsymbol{\Sigma}^{(k)}$ and K of the GMM in Eq. (3) can be estimated using a set of stationary samples generated from the unnormalized pdf $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. These samples can be easily generated using an MCMC sampler [25]. For example, random-walk Metropolis (RWM) [24] and its variants are best suited to sample from unimodal pdfs while transitional MCMC (TMCMC) [26] can sample from multimodal pdfs. Note that this MCMC sampling needs to be executed only once for the purpose of sparse learning.

Vast literature exists in the machine learning practice on ways to estimate the GMM parameters following the availability of training data (in this case, stationary MCMC samples). Kernel density estimation (KDE) approach is a rudimentary but quick way to construct a GMM for $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ as its parameters $a^{(k)}, \boldsymbol{\mu}_\alpha^{(k)}, \boldsymbol{\Sigma}^{(k)}$ and K are exactly known given the stationary samples [27]. Expectation-maximization (EM) algorithm is the preferred algorithm in machine learning for building GMMs with minimum number (K) of kernels [22]. Pettit and Wilson [28] demonstrated the applicability of variational Bayesian inference as an alternative to EM algorithm for the case of limited training data. Also, a Monte Carlo based approximation of $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ can be considered as a special type of GMM where the Gaussian kernels are reduced to Dirac-delta functions centered at individual $\boldsymbol{\phi}$ values. In this case, evidence from Eq. (7) reduces to the average of ARD prior $\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})$ values computed at stationary $\boldsymbol{\phi}$ samples from $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. Although computationally efficient, such an approach needs to be carefully adjusted for cases when samples from $\text{p}(\mathcal{D}|\boldsymbol{\phi})\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ are located far from the prior space $\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})$, which in turn varies with changing $\boldsymbol{\alpha}$ value.

In this work, we employ the KDE approach to build the GMM in Eq. (3) since the model dimensionality considered in the numerical investigations is manageable. However, we recommend using EM or the variational Bayesian inference approach for large model dimensionality (N_ϕ) cases to ensure the computational

efficiency of NSBL since the number of kernels K is significantly lower than the number of stationary samples used to estimate the GMM. However, any inaccuracies in GMM approximation resulting from choosing K value lower than necessary could result in erroneous sparsity levels and, therefore, adequate care should be taken in deciding the kernel size K. Nevertheless, NSBL is independent of the algorithm used to build the GMM for $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$, and therefore we leave it to the end-user to choose an appropriate algorithm for GMM construction. This paper is primarily aimed at introducing the analytical apparatus of NSBL, and the performance characteristic of different GMM building algorithms will be investigated in future studies.

The second numerical task involved in NSBL is the optimization of $\mathcal{L}(\log \boldsymbol{\alpha})$ in Eq. (13). Due to the unconstrained, non-convex nature of the optimization [23], we pursue a multistart Newton's method to estimate $\boldsymbol{\alpha}^{\text{map}}$ in Eq. (13). Newton's method operates by generating a sequence of iterates $\{\log \boldsymbol{\alpha}_i\}$ using the gradient vector $\mathbf{J}(\log \boldsymbol{\alpha}_i)$ and the Hessian matrix $\mathbf{H}(\log \boldsymbol{\alpha}_i)$ from Eq. (19) and Eq. (21), respectively. Denoting $\mathcal{L}(\log \boldsymbol{\alpha}_j) = \mathcal{L}_j$, $\mathbf{J}(\log \boldsymbol{\alpha}_j) = \mathbf{J}_j$ and $\mathbf{H}(\log \boldsymbol{\alpha}_j) = \mathbf{H}_j$, a Newton's iteration to obtain the new iterate $\boldsymbol{\alpha}_{j+1}$ is written as

$$\log \boldsymbol{\alpha}_{j+1} = \log \boldsymbol{\alpha}_j + \beta_j \mathbf{p}_j \ , \ \text{where} \ \ \mathbf{H}_j \mathbf{p}_j = -\mathbf{J}_j \tag{24}$$

and β_j is the step-length determined by satisfying Wolfe, Goldstein, or Armijo backtracking conditions [29].

Nocedal and Wright [29] provide a detailed discussion on many variants of Newton's algorithm designed for solving non-convex optimization problems where the Hessian is not guaranteed to be a positive definite matrix. Most of these variants of Newton's method fall under two categories:

1. Modified Newton method [29]: Instead of solving $\mathbf{H}_j \mathbf{p}_j = -\mathbf{J}_j$, the modified Newton method solves $(\mathbf{H}_j + \mathbf{E}_j)\mathbf{p}_j = -\mathbf{J}_i$ where $\mathbf{E}_j$ is an appropriate matrix added to the Hessian matrix $\mathbf{H}_j$ to make it positive definite.
2. Trust-region Newton method: The trust-region approach relies on building a quadratic approximation of $\mathcal{L}(\log \boldsymbol{\alpha})$ around the current iterate $\log \boldsymbol{\alpha}_i$, and determining an appropriate search direction $\mathbf{p}_j$ by solving a constrained optimization subproblem

$$\min \left(\mathcal{L}_j + \mathbf{J}_j^T \mathbf{p} + \frac{1}{2} \mathbf{p}^T \mathbf{H}_j \mathbf{p} \right) \quad \text{such that} \quad ||\mathbf{p}|| \leq \Delta_j \tag{25}$$

   where Δ_j is the trust-region radius [30]. See Conn *et al.* [30] for a detailed review of trust-region methods.

In general, both these algorithms should perform similarly given a reasonable choice of the additive matrix $\mathbf{E}_j$ in the modified Newton method. In this work, we opt for trust-region Newton method so as to avoid this additional step of choosing $\mathbf{E}_j$ (which is often problem specific). Nevertheless, any non-convex optimizer that can exploit the readily-available gradient and Hessian information is well-suited for NSBL. We will use the Scipy library [31] implementation of the trust-region algorithms to execute the Newton iteration. The resulting algorithm of NSBL is summarized in Algorithm 1.

2.7. Relation to SBL

In this section, we demonstrate the relationship between NSBL and SBL (or RVM) for the case of linear regression models and Gaussian errors. Consider the model $\mathbf{d} = \boldsymbol{\Psi}\boldsymbol{\phi} + \boldsymbol{\epsilon}$, where $\boldsymbol{\phi} \in \mathbb{R}^{N_\phi}$ is the unknown coefficient vector, $\mathcal{D} \equiv \mathbf{d} \in \mathbb{R}^{N_d \times 1}$ is the measurement vector, $\boldsymbol{\Psi} \in \mathbb{R}^{N_d \times N_\phi}$ is the design matrix, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \rho^{-1}\mathbf{I}_{N_d})$ is the model error where ρ is the error precision. SBL operates by conditioning the sparse learning apparatus on ρ, and then estimating it iteratively following each SBL update in $\boldsymbol{\alpha}$. This explicit conditioning on ρ is not shown here for brevity. SBL treats all parameters in $\boldsymbol{\phi}$ as questionable, and so $N_\phi{=}N_\alpha$ and $\boldsymbol{\phi}{=}\boldsymbol{\phi}_\alpha$ in the NSBL setup outlined in Section 2.1. Consequently, $\boldsymbol{\phi}_{\text{-}\alpha}$ is a null vector and the prior pdf $\mathrm{p}(\boldsymbol{\phi}|\boldsymbol{\alpha})$ in Eq. (1) is just the ARD prior $\mathcal{N}(\boldsymbol{\phi}|\mathbf{0}, \mathbf{A}^{-1})$. Under these circumstances, the

Algorithm 1: NSBL algorithm

Decompose $\boldsymbol{\phi}$ as $\{\boldsymbol{\phi}_{\alpha}, \boldsymbol{\phi}_{\text{-}\alpha}\}$ and assign a known prior pdf $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ to *a priori* relevant parameters $\boldsymbol{\phi}_{\text{-}\alpha}$;
Build a Gaussian kernel-based approximation for $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ by estimating $a^{(k)}, \boldsymbol{\mu}_{\alpha}^{(k)}, \boldsymbol{\Sigma}^{(k)}$ in Eq. (3);
Choose a hyperprior $\mathcal{G}(\phi_i|r_i, s_i)$ for each $\phi_i \in \boldsymbol{\phi}_{\alpha}$ using Table 1;
Choose a starting hyperparameter value $\log \boldsymbol{\alpha}_0$; set j=0;
while *not converged* **do**
  Given $\boldsymbol{\alpha}_j$, compute $\mathbf{m}^{(k)}, \mathbf{P}^{(k)}, \mathbf{B}_{\alpha}^{(k)}$ using Eq. (B.4);
  Compute weight $w^{(k)}$ using Eq. (11) and factor $v_i^{(k)}$ using Eq. (18);
  Compute gradient vector $\mathbf{J}(\log \boldsymbol{\alpha}_j)$ using Eq. (19);
  Compute Hessian matrix $\mathbf{H}(\log \boldsymbol{\alpha}_j)$ using Eq. (21);
  Compute the new iterate $\log \boldsymbol{\alpha}_{j+1}$ using Newton's iteration as per Eq. (24);
  Compute relevance indicator γ_i^{rms} using Eq. (23);
  Set j=j+1;
end

approximation for $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})$ in Eq. (3) is known analytically from maximum likelihood estimation (MLE) or ordinary least-square (OLS) theory [22] as

$$\mathrm{p}(\mathcal{D}|\boldsymbol{\phi}) = \mathcal{N}(\mathbf{d}|\boldsymbol{\Psi}\boldsymbol{\phi}, \rho^{-1}\mathbf{I}_{N_d}) = \hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\phi}) = a^{(1)}\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)}) \tag{26}$$

where $\boldsymbol{\mu}^{(1)} = (\boldsymbol{\Psi}^T\boldsymbol{\Psi})^{-1}\boldsymbol{\Psi}^T\mathbf{d}$ is the MLE/OLS estimate, and $\boldsymbol{\Sigma}^{(1)} = (\rho\boldsymbol{\Psi}^T\boldsymbol{\Psi})^{-1}$ is the covariance matrix of the MLE estimate $\boldsymbol{\mu}^{(1)}$. Note that the sampling distribution $\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(1)}, \boldsymbol{\Sigma}^{(1)})$ produced from MLE/OLS is same as the posterior pdf obtained using Bayesian linear regression with no or flat priors [32]. Given $\boldsymbol{\mu}^{(1)}$ and $\boldsymbol{\Sigma}^{(1)}$, the coefficient $a^{(1)}$ in also available analytically from Eq. (26).

The model evidence from Eq. (9) is obtained as $\mathrm{p}(\mathcal{D}|\boldsymbol{\alpha})$=$\mathcal{N}(\mathbf{d}|\mathbf{0}, \boldsymbol{\Psi}\mathbf{A}^{-1}\boldsymbol{\Psi}^T + \rho\mathbf{I}_{N_\phi})$. Given the single kernel (K=1) representation of the likelihood function in Eq. (26), the only weight coefficient $w^{(1)}$ in Eq. (11) is equal to one. The posterior pdf from Eq. (12) is available as $\mathrm{p}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha})$=$\mathcal{N}(\boldsymbol{\phi}|\mathbf{m}^{(1)}, \mathbf{P}^{(1)})$, where $\mathbf{P}^{(1)}$=$(\rho\boldsymbol{\Psi}^T\boldsymbol{\Psi} + \mathbf{A})^{-1}$ and $\mathbf{m}^{(1)}$=$\rho\mathbf{P}^{(1)}\boldsymbol{\Psi}^T\mathbf{d}$ from Eq. (B.4). The gradient $J_i(\log \boldsymbol{\alpha})$ of $\mathcal{L}(\log \boldsymbol{\alpha})$ in Eq. (19a) is obtained as

$$J_i(\log \boldsymbol{\alpha}) = \frac{\gamma_i^{(1)} - \alpha_i(m_i^{(1)})^2}{2} + r_i - s_i\alpha_i \tag{27}$$

Setting this gradient $J_i(\log \boldsymbol{\alpha})$ to zero leads to

$$\alpha_i = \frac{\gamma_i^{(1)} + 2r_i}{(m_i^{(1)})^2 + 2s_i} \tag{28}$$

This solution to α_i is exactly the expression exploited in SBL to perform iterative re-estimation of α_i [14].

In summary, NSBL and SBL possess the same sparsity-inducing Bayesian apparatus for linear regression models; the only difference being the way the model evidence or $\mathcal{L}(\log \boldsymbol{\alpha})$ is optimized. SBL operates by setting the gradient with respect to α_i (Eq. (27)) to zero, while NSBL exploits the gradient and Hessian information to execute a multistart Newton iteration. Notice that while SBL updates only a single α_i per iteration, NSBL updates the entire hyperparameter vector $\boldsymbol{\alpha}$ at each Newton iteration. In Section 3.1 we explore how this difference affects the numerical performance of the two algorithms. Nevertheless, NSBL is applicable to nonlinear physics-based models, while SBL is designed for data-based modelling involving linear-in-parameter models (regression or classification).

3. Numerical investigations

3.1. Linear regression: Sparse PCE expansion of Ishigami function

In this section, we consider a linear regression testbed for investigating the performance of NSBL in comparison with SBL and BCS. Since NSBL is similar to SBL for a linear regression setting (as illustrated in Section 2.7), we expect similar sparsity levels from both the algorithms. We acknowledge that the conclusions from this exercise may not be transferable to physics-based sparse learning where the models are typically in the form of nonlinear, stochastic differential equations. This exercise is also intended to contrast the efficiency of these algorithms in terms of the number of evidence computations required to reach the optimal sparse solution.

The linear-in-parameter model considered here is a Polynomial Chaos Expansion (PCE) surrogate. PCE surrogates have become omnipresent in computational physics applications as a cheap replacement for high-fidelity, time-consuming numerical solvers for the forward propagation of uncertainties. As reported in Appendix C, a slight increase in PCE order p or the dimension d creates a tremendous increase in the number of PCE terms. Estimation of these large numbers of PCE coefficients demands a proportionally large number of model evaluations from time-consuming computer codes. This creates a computational bottleneck commonly known as the *curse of dimensionality*. Since the PCE coefficients are inherently sparse (only a few PCE terms are consequential), this computational bottleneck is remedied by seeking a sparse PCE representation. See Ghanem [33] for a comprehensive text on PCEs.

Ishigami function is a popular test bed for benchmarking sparse PCE construction algorithms due to its nonlinear and non-monotonic behaviour with respect to input variables [34]. Ishigami function is written as [34]

$$Y = \sin X_1 + a \sin^2 X_2 + b X_3^4 \sin X_1 \tag{29}$$

where the input vector is $\mathbf{X} = \{X_1, X_2, X_3\}$ and the output quantity-of-interest is Y. Each input X_i is uniformly distributed within $[-\pi, \pi]$ and parameters $a = 7.0$ and $b = 0.1$ are known. Given the uniformly distributed germs ξ_i, Legendre polynomials are employed for constructing a PCE surrogate of the Ishigami function.

An inverse problem is posed by generating 250 samples of input $\mathbf{X} = \{X_1, X_2, X_3\}$ using Latin Hypercube Sampling (LHS) [34]. These samples are then pushed forward through the Ishigami function in Eq. (29) to generate the corresponding samples for output Y. Next, multiple PCE surrogates with varying orders are proposed to fit this data. The order of PCE surrogates is varied as $p = 1, 2, 3, 4, 5, 6, 7$. Notice that the number of PCE terms for a 7^{th}-order PCE surrogate will be $P = (7+3)!/(7!3!) = 120$. Next, NSBL, SBL, and BCS are employed to seek the sparse representation of PCE coefficients. The numerical implementation of SBL and BCS algorithms considered here involves iterative addition-deletion of basis, also known as fastSBL [15] and fastLAPLACE [16], respectively. More details regarding SBL and BCS are provided in Appendix A. For both SBL and BCS, we employ flat hyperpriors, and α_i at each iteration of fastSBL or fastLAPLACE is chosen in a deterministic fashion by iterating index i from zero to N_α.

The NSBL algorithm employed for this exercise is summarized in Algorithm 1. Since PCE surrogates are linear-in-parameter and all the unknown parameters (PCE coefficients) are assigned ARD priors, the GMM approximation in Eq. (3) is known analytically, as detailed in Section 2.7. Hence, the number of Gaussian kernels in Eq. (3) is one, and the mean and covariance of this kernel are known using MLE theory. Furthermore. the sparsity levels produced from NSBL are determined using the relevance indicator γ_i^{rms} defined in Section 2.5. Three different values will be chosen for tolerance γ^{tol} for implementing NSBL, wherein $\gamma_i < \gamma^{\text{tol}}$ will imply the corresponding PCE basis is irrelevant. Using Table 1, hyperprior for α_i in NSBL is chosen to be flat in log-space, i.e. $\mathrm{p}(\log \alpha_i) \propto 1$ by assigning r_i=s_i=1e-05. The model error variance ρ^{-1} is estimated in similar fashion as in SBL and BCS as $(N_d - \sum \gamma_i + 2a)/(||\mathbf{d} - \mathbf{\Psi m}||^2 + 2b)$ where $\mathbf{m}$ is the posterior mean. A large starting value of $\log \alpha_i = 5.0$ is chosen to ensure all PCE basis are absent from the surrogate at the start of the NSBL algorithm. The mean coefficient is set at $\log \alpha_0 = 0.0$ to ensure a non-zero mean of Y is captured in the beginning.

Figure 2a shows the index-of-sparsity identified by SBL, BCS and NSBL. The index of sparsity is defined as the ratio of relevant coefficients with the total number of coefficients (N_ϕ or N_α). For example, SBL

identified 45 relevant PCE basis for the sixth-order PCE surrogate having a total of 84 terms, resulting in an index-of-sparsity of 0.536. As evident from Figure 2a, BCS produces sparser solution than SBL. This increased sparsity in BCS is at the expense of lower model evidence at the optimum (as seen in Figure 2c). This fact about SBL vs BCS has been previously proven through analytical means [16]. The index-of-sparsity for NSBL is reported for relevance indicator tolerance γ^{tol} set at 0.25, 0.50 and 0.75. For instance, when $\gamma^{\text{tol}} = 0.50$, any questionable parameter having the relevance indicator γ_i^{rms} value less than 0.5 is deemed irrelevant.

As evident from Figure 2a, the index-of-sparsity from NSBL with γ^{tol}=0.25 resembles closely to those obtained using SBL. This observation provides reassurance in NSBL methodology as both NSBL and SBL are optimizing the same cost function (as per Section 2.7). Also, as γ^{tol} increases, NSBL tends to produce a sparser solution. NSBL with $\gamma^{\text{tol}} = 0.75$ results in similar sparsity levels as BCS. Also, the index-of-sparsity decreases with increasing PCE order (or number of PCE basis). This observation implies that the addition of a higher-order PCE basis does not contribute much to the understanding of the Ishigami function, and most of these PCE bases end up being irrelevant.

Figure 2b reports the number of iteration counts or the number of times the model evidence is computed to reach the optimum. NSBL involves significantly less evidence count since each iteration updates with entire vector $\boldsymbol{\alpha}$ using analytically available gradient and Hessian information. On the contrary, SBL and BCS iterate through individual α_i's and exploit only the gradient information, leading to a higher iteration count to reach the optimum. Figure 2d shows the model error variance pertaining to the optimum. The model error variance decreases continuously with increasing PCE order as the models are getting better, even if they are getting sparser (as per Figure 2a). The model error variance stagnates beyond sixth-order PCE as no significant knowledge remains to be gained regarding the Ishigami function. Note that the results reported in Figure 2 were generated for multiple instances of 250 $\mathbf{X} - Y$ samples generated using LHS. Although the index-of-sparsity identified by SBL, BCS, and NSBL were slightly different, the relative trend in index-of-sparsity, optimal model evidence, and model error variance were observed to be similar to those reported in Figure 2. Also, the kernel-based computation of gradient and Hessian of model evidence from NSBL was validated using finite-difference for varying PCE order and $\log \alpha$ values.

The Sobol sensitivity indices pertaining to the sparse representation of the seventh-order PCE surrogate were within 1% error of the analytical values (results not reported here). In fact, even when using only 50 LHS samples, the sensitivity indices obtained using sparse PCE were within 5% error of the analytical values. This is possible since only 42 PCE terms in seventh-order PCE were relevant. In summary, the similarity in results validates the proposed algorithm of NSBL against SBL for a linear regression setting. In addition, it is shown that NSBL requires a lesser number of evidence computations than SBL since it exploits the Hessian information of model evidence to expedite the evidence maximization.

3.2. Polynomial regression with a multimodal prior pdf

We consider a polynomial regression example wherein a multimodal prior pdf is assigned to an a priori relevant parameter, which then induces multimodality in the posterior pdf and the model evidence. We investigate the applicability of NSBL under such adverse circumstances where SBL and BCS are inapplicable. This example demonstrates the performance of NSBL for physics-based inverse problems where multimodality exists in either the parameter space or the hyperparameter space.

Figure 3 shows the noisy observational data generated using the polynomial $y_i = 1 + x_i^2 + \epsilon_i$, where ϵ_i is a Gaussian white noise process with pdf $\mathcal{N}(\epsilon_i|0, 0.02)$. The observations consist of 50 noisy samples equally spaced in $x \in [0.75, 1.25]$. An inverse problem is posed to understand the truth ($y = 1 + x^2$) by using these observations. To mimic the practical circumstances, an over-parameterized polynomial $y{=}a_0{+}a_1x{+}a_2x^2{+}\epsilon$ is proposed to model the observations. Subsequently, sparsity in $\boldsymbol{\phi} = \{a_0, a_1, a_2\}$ needs to be identified. The observational noise ϵ is assumed as known ($\mathcal{N}(\epsilon|0, 0.02)$) for the sake of simplicity. Although the noise strength could also be estimated within the proposed NSBL setup, this is not considered necessary to highlight the usefulness of NSBL for this numerical exercise.

To mimic the physics-based modelling circumstances, parameter a_0 is treated as a priori relevant. As a result, $\boldsymbol{\phi}$ is decomposed into $\boldsymbol{\phi}_\alpha = \{a_1, a_2\}$ and $\boldsymbol{\phi}_{\text{-}\alpha} = \{a_0\}$. The known prior pdf pf a_0 is chosen by realizing

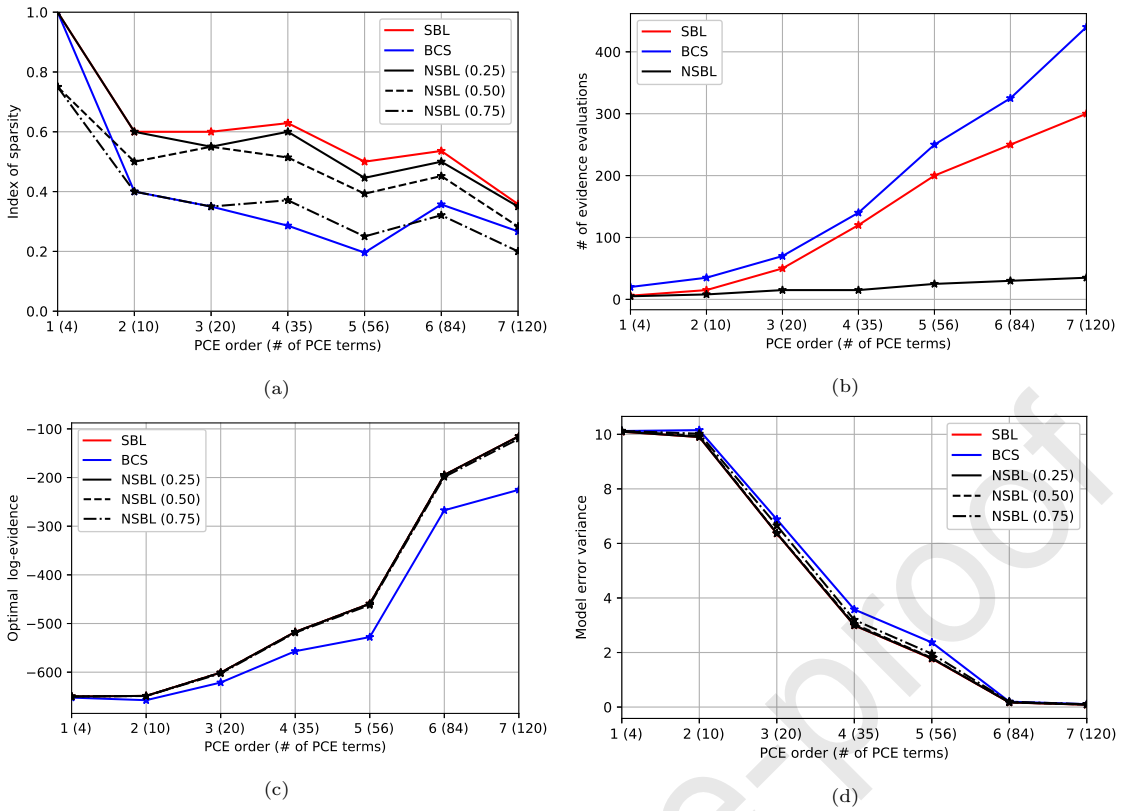


Figure 2: Variation of a) index of sparsity, b) number of model evaluations (iteration count), c) optimal log-evidence, and d) model error variance with increasing PCE order. The sparsity levels produced from NSBL algorithm pertains to γ_i tolerance set at 0.25, 0.50 and 0.75. A γ_i value lower than this tolerance value means the corresponding PCE basis is considered to be irrelevant.

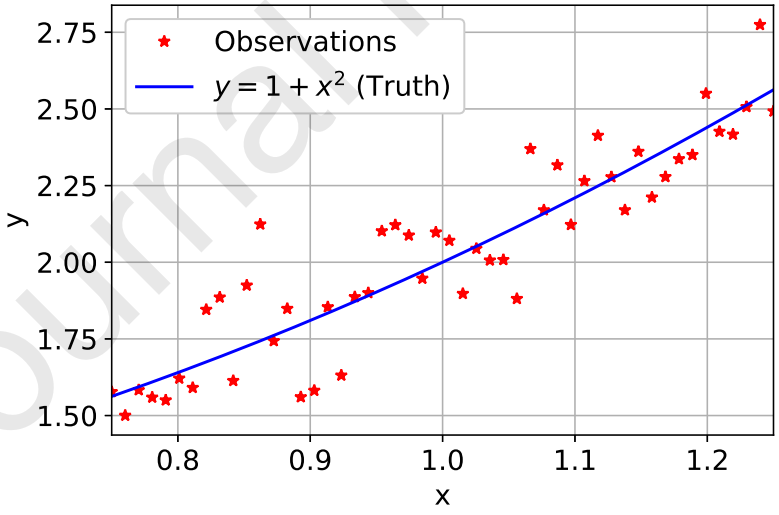
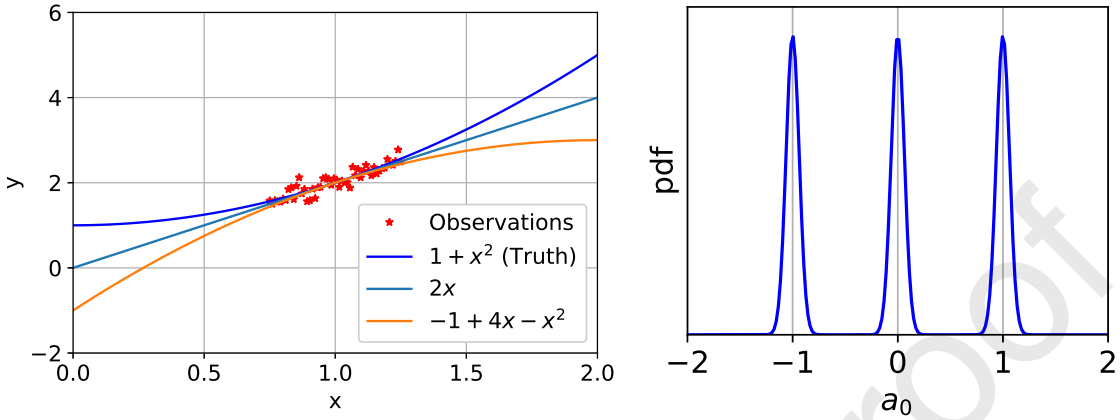


Figure 3: Noisy observations versus the truth

(using Figure 4a) that the following three nested models demonstrate a reasonable fit to the observations: 1) $y = 2x$, 2) $y = 1 + x^2$, and 3) $y = -1 + 4x - x^2$. Using this information, the prior pdf for a_0 is chosen as a mixture of equally-weighted Gaussian kernels centered at minus one, zero, and one, as shown in Figure 4b. The location of these kernels demonstrate our prior belief about the potential fit of the three nested models. Note that one of these modes ($a_0 = 1$) also represent the truth ($y = 1 + x^2$).



(a) Possible data-fit scenarios

(b) Prior pdf for a_0

Figure 4: NSBL setup

Next, we employ NSBL to investigate the sparse structure of questionable parameter vector $\boldsymbol{\phi}_\alpha = \{a_1, a_2\}$ of the quadratic polynomial $y{=}a_0 + a_1 x + a_2 x^2 + \epsilon$. This problem set-up is summarized in Table 3. ARD priors are assigned to a_1 and a_2 with precision α_1 and α_2, respectively. Notice that the likelihood function and the posterior pdf are a three-dimensional function of $\boldsymbol{\phi}$, while the model evidence is a two-dimensional function of $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2\}$. Also, the hyperprior for each $\alpha_i \in \boldsymbol{\alpha}$ is chosen to be uninformative by assigning $\log r_i = \log s_i$ =-10 in Eq. (6). In this case, the optimum of objective function $\mathcal{L}(\log \boldsymbol{\alpha})$ is entirely dictated by the log-evidence estimator $\log \hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha})$ from Eq. (9).

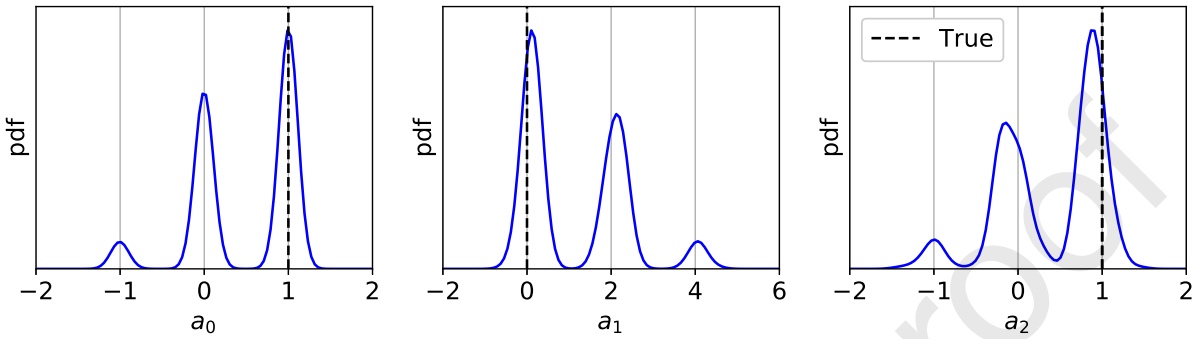
Proposed model	$y = a_0 + a_1 x + a_2 x^2 + \epsilon \;\; ; \;\; \epsilon \sim \mathcal{N}(0, 0.02)$
$\boldsymbol{\phi}$ decomposition	$\boldsymbol{\phi}_\alpha = \{a_1, a_2\} \;\; , \;\; \boldsymbol{\phi}_{\text{-}\alpha} = \{a_0\}$
Known prior, $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$	$\{\mathcal{N}(a_0\text{-}1, 0.2^2) + \mathcal{N}(a_0\|0, 0.2^2) + \mathcal{N}(a_0\|1, 0.2^2)\}/3$
ARD prior, $\mathrm{p}(\boldsymbol{\phi}_\alpha\|\alpha)$	$\mathcal{N}(a_1\|0, \alpha_1^{-1})\; \mathcal{N}(a_2\|0, \alpha_2^{-1})$
Hyperprior, $\mathrm{p}(\boldsymbol{\alpha})$	$\prod_{i=1}^{2} \mathcal{G}(\alpha_i\|r, s)$; $\log r = \log s =$ -10

Table 3: NSBL setup for the polynomial regression model with a multimodal prior pdf for a_0.

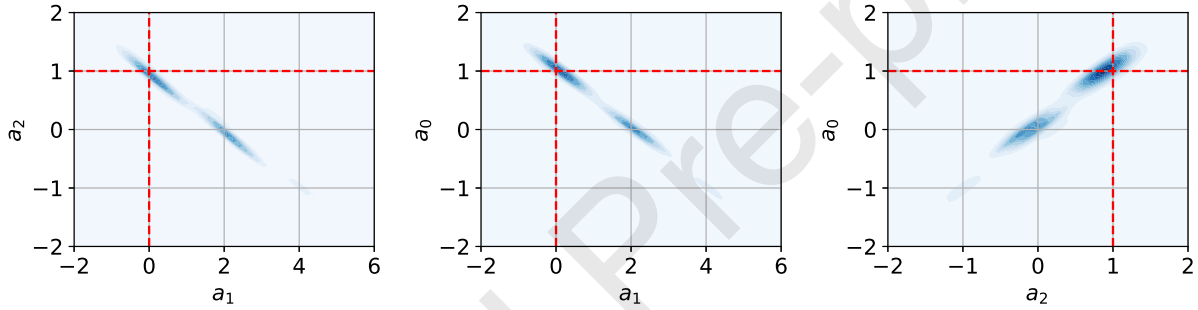
Initiating NSBL, TMCMC algorithm [26] is employed to generate 2500 stationary samples from the partial posterior pdf $\propto \mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$. The TMCMC algorithm is well-suited to generate iid samples from multimodal and/or high-dimensional pdfs with minimum manual intervention [26]. Given these iid samples, a multivariate KDE approximation of the partial posterior is constructed. This KDE-based GMM of partial posterior involves 2500 equally-weighted Gaussian kernels centered at individual TMCMC sample locations and having the same covariance. The covariance matrix of the KDE kernels is equal to the sample covariance of 2500 TMCMC samples, scaled by a factor computed using Scott's rule [35]. The multivariate KDE approximation is constructed using the Scipy library's *gaussian_kde* function [31].

Figure 5a show the marginal parameter pdfs, and figure 5b shows the joint parameter pdfs, both pertain-

ing to the KDE approximation of $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ using Gaussian kernels. In Figure 5a, the partial posterior pdf is peaked highest at the true value of $\boldsymbol{\phi} = \{1, 0, 1\}$ (or model $y = 1+x^2$). The other two possibilities of $\boldsymbol{\phi} = \{0, 2, 0\}$ (or model $y = 2x$) and $\boldsymbol{\phi} = \{-1, 4, -1\}$ (or model $y = -1+4x-x^2$) possess a lower yet non-zero posterior probability. This behavior of $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ is indicative of the irrelevance of a_1 (value of zero) and the relevance of a_2 (value of one). However, determining parameter relevance by examining posterior pdfs is an impractical and arbitrary process for high-dimensional physics-based models. As we demonstrate next, the evidence-based NSBL framework provides an efficient quantitative alternative to identify sparsity while dealing with non-Gaussian posterior pdfs.



(a) Marginal parameter pdfs.



(b) Joint parameter pdfs.

Figure 5: GMM representation for $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ using multivariate KDE approximation.

The next step of NSBL algorithm involves executing the Newton iteration to compute the optimal hyperparameter vector $\boldsymbol{\alpha}^{\text{map}}$ by maximizing the objective function $\mathcal{L}(\log \boldsymbol{\alpha})$ from Eq. (15). Figure 6 show the NSBL results when initiating the Newton iteration from $\log \boldsymbol{\alpha} = \{6, 8\}$. In Figure 6a, the optimal hyperparameter vector is computed as $\log \boldsymbol{\alpha}^{\text{map}} = \{-1.42, 6.76\}$ in only ten Newton iterations. In Figure 6b, the relevance indicator for a_1 approaches zero while that of a_2 approaches a value of one, thereby indicating the irrelevance of a_2 and the relevance of a_1. This optimum pertains to the nested model of $y = 2x$. The resulting posterior pdf $\mathrm{p}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha}^{\text{map}})$ is also available analytically and is shown in Figure 6c.

Figure 7 shows the NSBL results when initiating from $\log \boldsymbol{\alpha} = \{-3, -3\}$. The optimal hyperparameter vector is computed as $\log \boldsymbol{\alpha}^{\text{map}} = \{5.24, 0.00\}$. In Figure 7b, the relevance indicator for a_1 converges to one, while that for a_2 converges to zero. This demonstrates the irrelevance of a_1 and the relevance of a_2. Based on Figure 7a, this optimum is the global optimum, and the sparse model thus identified is the true model $(y = 1 + x^2)$. In Figure 7c, the posterior pdf for a_1 following the sparse learning approaches a Dirac-delta function centered at zero, indicating the removal of a_1 from the proposed model. On the other hand, the posterior parameter pdf of relevant parameters a_0 and a_2 is a uniquely-peaked Gaussian pdf centered around the true values.

The results reported in Figure 6 and Figure 7 brings out two key points. First, multimodality in log-

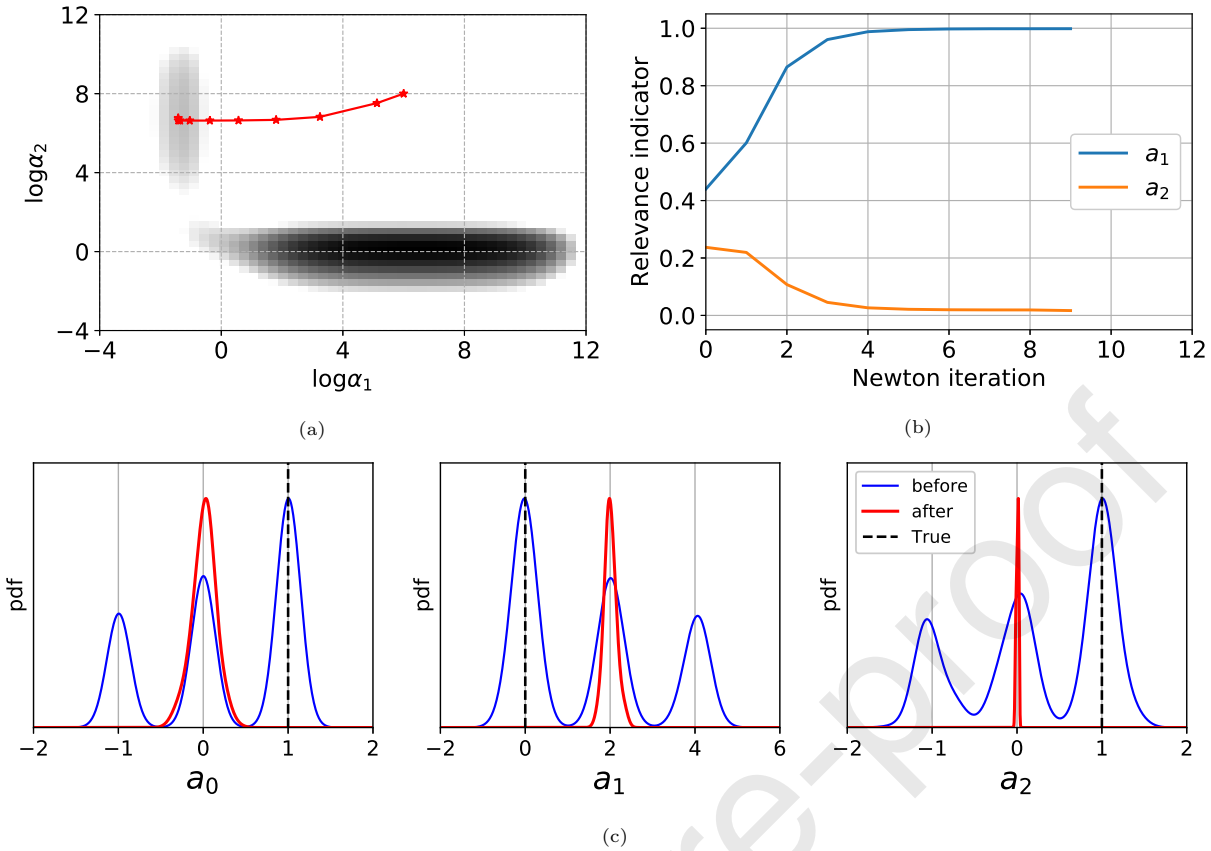


Figure 6: NSBL results when initiating from $\boldsymbol{\alpha} = \{-6, -8\}$: a) $\mathcal{L}(\log \boldsymbol{\alpha})$ iterates in relation to the actual $\mathcal{L}(\log \boldsymbol{\alpha})$, b) variation of relevance indicator during Newton iteration, c) marginal posterior pdfs before and after the inclusion of optimal ARD priors.

evidence or $\mathcal{L}(\log \boldsymbol{\alpha})$ can be induced by multimodality in the prior pdf. To our best knowledge, multimodality in model evidence has not been studied or reported previously in the scientific literature. This example demonstrates the need for cautious use of evidence-based model selection tasks in the presence of multimodal prior pdfs. Second, a multistart of the Newton iteration is necessary for the convergence of NSBL to the global optimum.

The benefit of sparse learning can be realized by contrasting the extrapolated response from the sparse model with that of the proposed model. Figure 8a shows prediction made using the posterior parameter pdf ($\propto \mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$) while using flat priors for questionable parameters, Figure 8b shows the prediction made using the sparse model based on the local evidence optima of $\log \boldsymbol{\alpha} = \{-1.42, 6.76\}$(Figure 6), and Figure 8c shows the prediction made using the sparse model based on the global optimum of $\log \boldsymbol{\alpha} = \{6.24, -4.00\}$ (Figure 7). In Figure 8a, using flat prior pdfs for questionable parameters lead to multimodality and a large variance in the predictions. The use of flat priors has been the engineering practice under the lack of prior knowledge. This approach is problematic in this case. NSBL thus offers a much more principled alternative to assigning flat priors using ARD. As shown in Figure 8c , the extrapolated response from the sparse model is a drastic improvement over the use of the over-parameterized quadratic model with flat priors.

This numerical example demonstrates the applicability of NSBL as a robust sparse learning tool while dealing with multimodality in the posterior parameter space or the hyperparameter space. Also, NSBL allows for the inclusion of prior knowledge in the form of non-Gaussian prior pdfs, making it well-suited for physics-based applications. Reiterating, the inclusion of prior knowledge and the handling of multimodal posterior pdfs are two key benefits of NSBL over SBL and BCS.

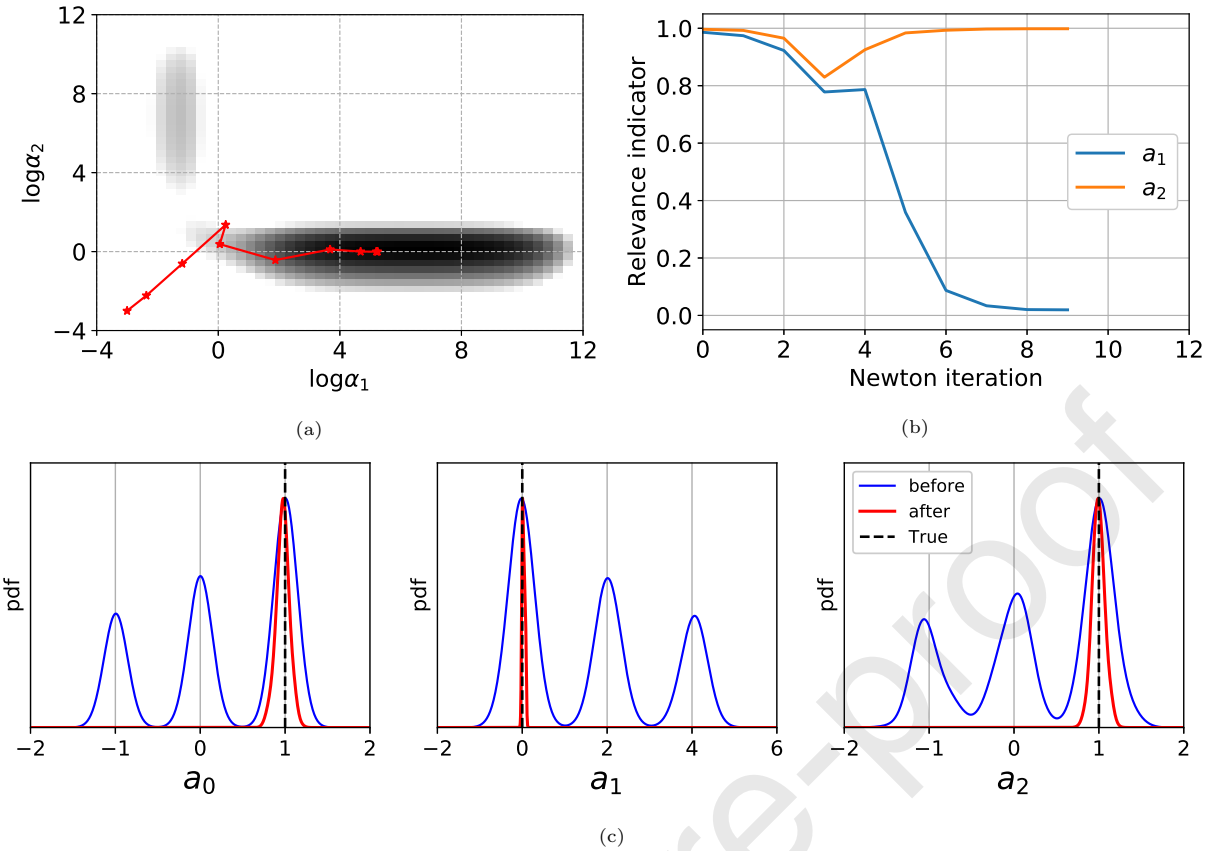


Figure 7: NSBL results when initiating from $\boldsymbol{\alpha} = \{-3, -3\}$: a) $\mathcal{L}(\log \boldsymbol{\alpha})$ iterates in relation to the actual $\mathcal{L}(\log \boldsymbol{\alpha})$, b) variation of relevance indicator during Newton iteration, c) marginal posterior pdfs following the inclusion of optimal ARD priors.

3.3. Three-dimensional mass-spring-damper system

In this section we consider a multi-storey shear building frame with rigid floors being modeled as a three-dof mass-spring-damper system, as shown in Figure 9a. Given the displacement vector $\mathbf{u} = \{u_1, u_2, u_3\}$, the equation-of-motion for the mass-spring-damper system is obtained as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f}(t), \tag{30}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the damping matrix, $\mathbf{K}$ is the stiffness matrix, and $\mathbf{f}(t)$ is the external forcing. For the current numerical study we will consider the case of free-vibration, i.e. $\mathbf{f}(t) = 0$. Given the state vector $\mathbf{x} = \{\mathbf{u}, \dot{\mathbf{u}}\}$, Eq. (30) is represented in the state-space form as $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$, where matrix $\mathbf{A} = [\mathbf{0}, \mathbf{I}; -\mathbf{M}^{-1}\mathbf{K}, -\mathbf{M}^{-1}\mathbf{C}]$. Given the initial state $\mathbf{x}_0$, the solution to the linear first-order differential equation $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$ is obtained as $\mathbf{x}(t) = e^{At}\mathbf{x}_0$. Following values are used for simulating the three-dof system for this study: $\mathbf{x}_0 = \{0, 1, 0, 0, 0, 0\}$, $m_1 = m_2 = m_3 = 1.0$, $k_1 = k_2 = k_3 = 1000.0$, $c_1 = 10$, $c_2 = 0$, $c_3 = 0$. Notice that the damping is only present between the first and the second storey. Figure 9b shows a noisy time-history of the third-floor (u_3) displacement obtained by corrupting the simulated (true) response with Gaussian white noise process that has a zero mean and a variance of 0.1. The observations consist of 100 points spread over four seconds. The observations are deemed incomplete since the displacement at the first and second floor is not observed.

Given the observations in Figure 9b, we pose an inverse problem to estimate the interstorey damping and stiffness coefficients. For simplicity, the mass ($m_i = 1.0$) and the observational noise strength are assumed to be known. To imitate the real-life circumstances, we assume that we have no prior knowledge

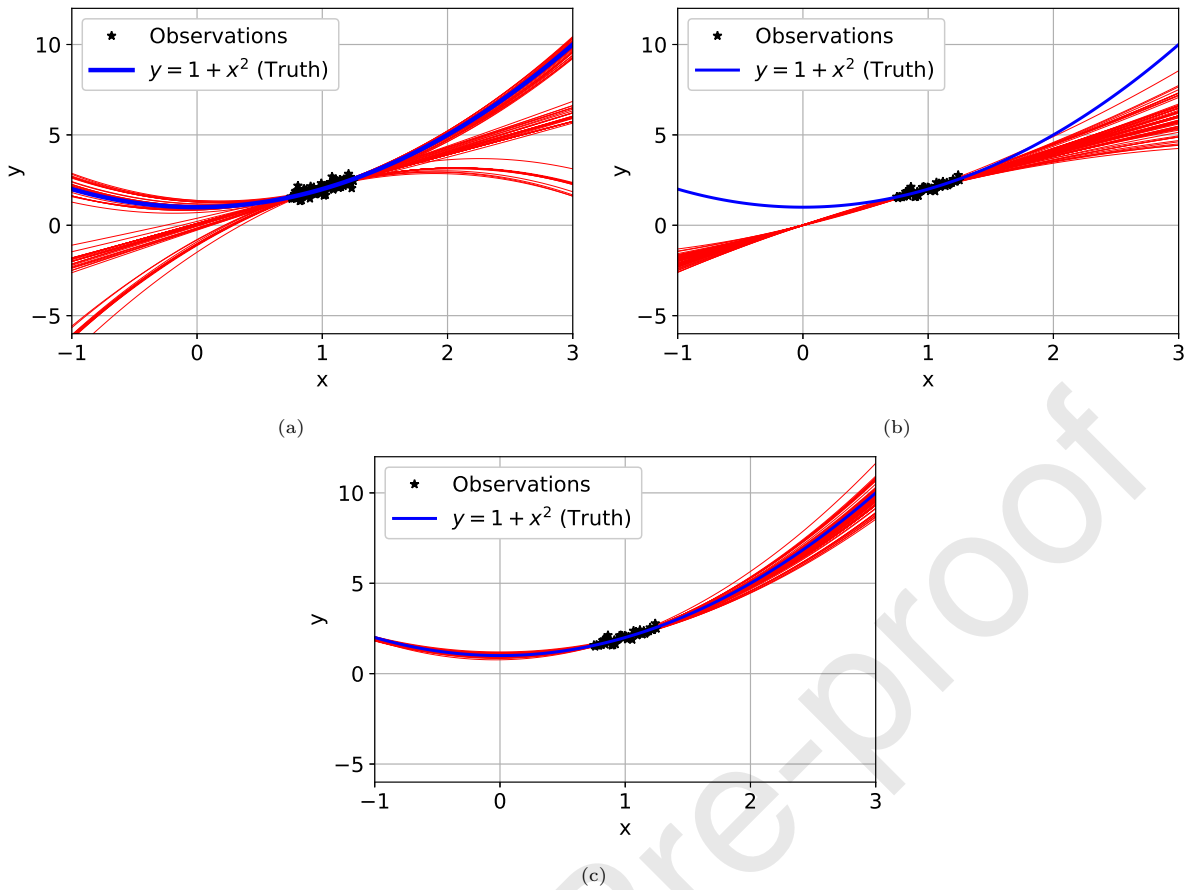


Figure 8: Model predictions using the $\log \boldsymbol{\alpha}$ value of a) $\{-\infty, -\infty\}$ (flat priors), b) $\{-1.42, 6.76\}$ (local optimum), and c) $\{6.24, -4.00\}$ (global optimum).

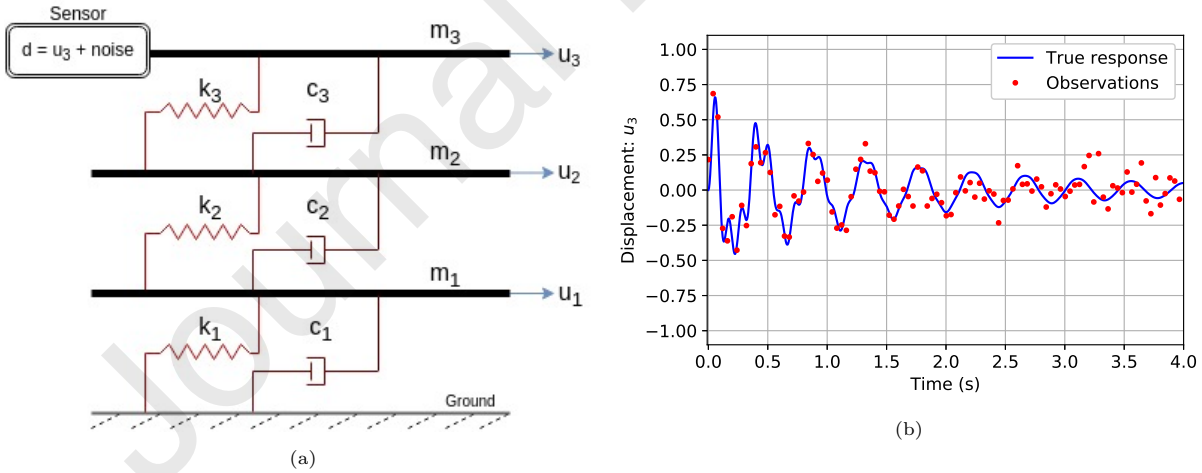


Figure 9: a) Three-dof mass-spring-damper model of a multi-storey building; b) Observed versus true response.

of the underlying damping representation. In other words, it is not known a priori that the damping is only active between the first and second storeys. Under such conditions, the damping is assumed to be

present in-between all floors. The unknown parameter vector for this model becomes $\boldsymbol{\phi} = \{c_1, c_2, c_3, k_1, k_2, k_3\}$. Given that stiffness coefficients are always positive, we assign a uniform pdf $\mathcal{U}(k_i|0, 5000)$ to all the three stiffness coefficients. Also, since we have no prior knowledge regarding damping coefficients, they are deemed questionable and are assigned ARD priors. In other words, $\boldsymbol{\phi}$ is decomposed into the questionable parameter vector $\boldsymbol{\phi}_{\alpha} = \{c_1, c_2, c_3\}$, and the a priori relevant parameter vector $\boldsymbol{\phi}_{\text{-}\alpha} = \{k_1, k_2, k_3\}$. Jeffrey's prior is used for ARD hyperparameters as per Table 1 such that the optima are solely dictated by model evidence. This NSBL setup is summarized in Table 4.

Proposed model	$\mathbf{M\ddot{u} + C\dot{u} + Ku = 0}$
$\boldsymbol{\phi}$ decomposition	$\boldsymbol{\phi}_{\alpha} = \{c_1, c_2, c_3\}$, $\boldsymbol{\phi}_{\text{-}\alpha} = \{k_1, k_2, k_3\}$
Known prior, $\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$	$\mathcal{U}(k_1\|0, 5000)\ \mathcal{U}(k_2\|0, 5000)\ \mathcal{U}(k_3\|0, 5000)$
ARD prior, $\mathrm{p}(\boldsymbol{\phi}_{\alpha}\|\boldsymbol{\alpha})$	$\mathcal{N}(c_1\|0, \alpha_1^{-1})\ \mathcal{N}(c_2\|0, \alpha_2^{-1})\ \mathcal{N}(c_3\|0, \alpha_3^{-1})$
Hyperprior, $\mathrm{p}(\boldsymbol{\alpha})$	$\prod_{i=1}^{3} \mathcal{G}(\alpha_i\|r, s)$; $\log r = \log s =$ -10

Table 4: NSBL setup for the three-dof mass-spring-damper model.

Initiating NSBL, TMCMC algorithm is exploited to generate 2500 iid samples from the partial posterior pdf $(\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha}))$. Subsequently, a multivariate KDE approximation using Gaussian kernels is constructed for $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ that consists of 2500 kernels, as per Eq. (3). The marginal pdfs pertaining to this KDE approximation are shown in Figure 10. Notice that although we are dealing with a linear structural dynamics model, the non-Gaussian nature of the partial posterior pdf is due to the nonlinear relation between the unknown parameters (damping and stiffness coefficients) and the observable entity (displacement at the third storey).

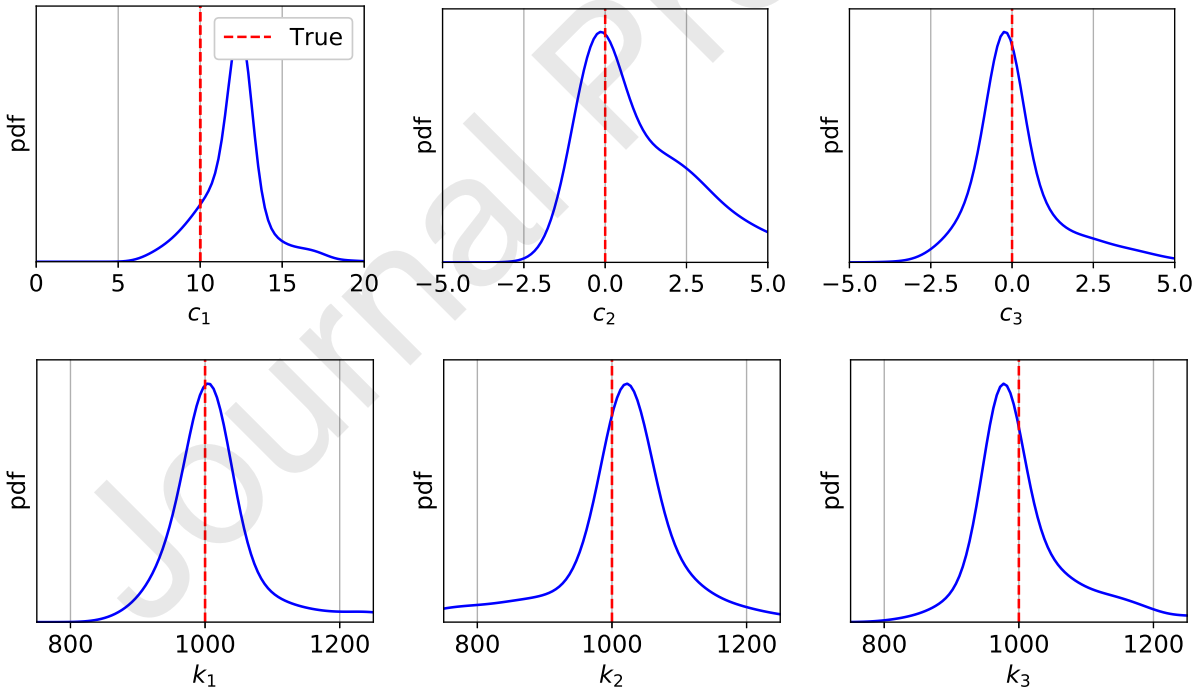


Figure 10: Marginal posterior pdf obtained using flat priors for damping coefficients and uniform prior $\mathcal{U}(0, 5000)$ for stiffness coefficients.

Although the GMM approximation of $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$ is centered around the true value (in Figure 10), the fitted model suffers from overfitting. One possible indication of overfitting is the non-zero posterior probability for negative values of the damping coefficients c_2 and c_3 in Figure 10. Since the structural damping is always positive, this behavior of posterior pdf indicates the non-physical aspect of the posterior pdf. Further, Figure 11 shows the predicted velocity response at the first storey, i.e. $\dot{u}_1$. The large variation in predictions is the result of overfitting produced by the over-parameterized model consisting of non-zero damping on all floors. NSBL is designed to handle such scenarios by sparsifying the parameter space to produce an optimally-parameterized model (in the sense of model evidence) that is free from overfitting.

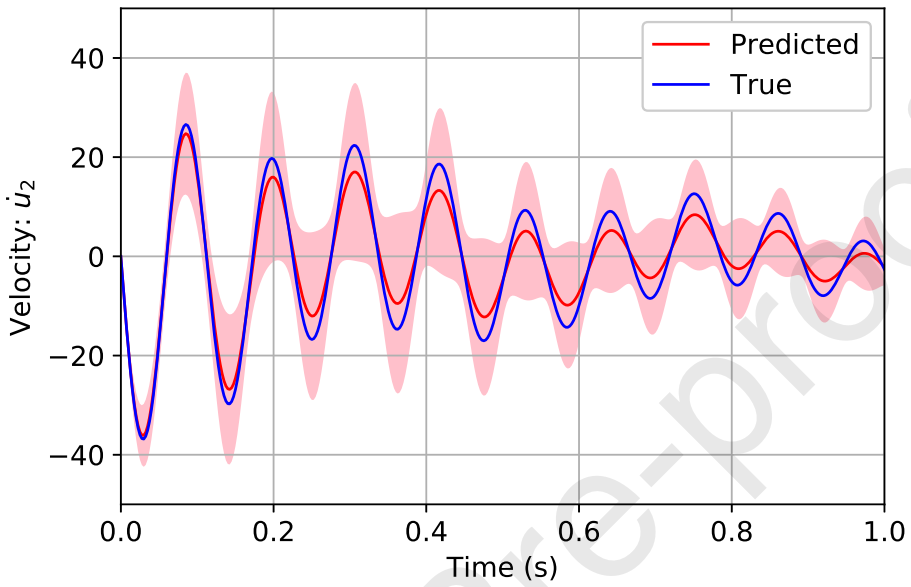
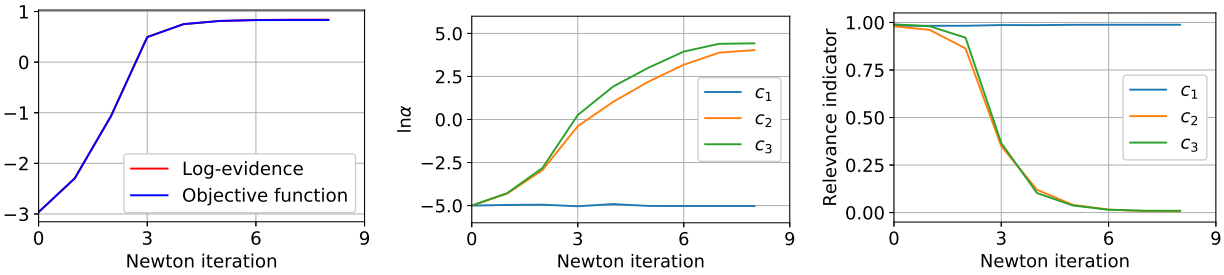


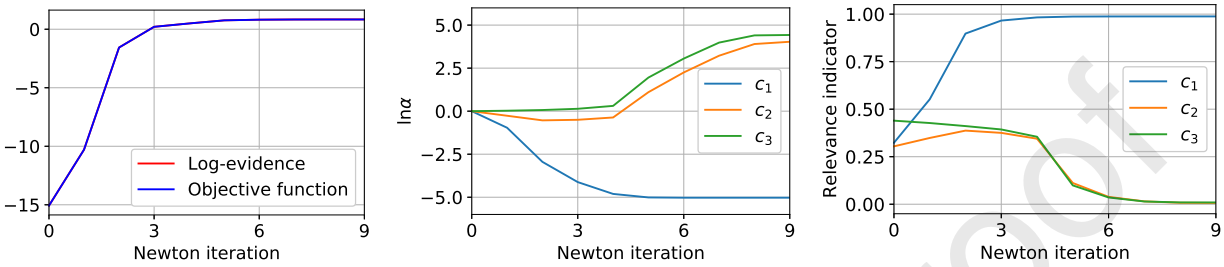
Figure 11: Predicted versus true time-history of velocity at first storey. The shaded area represents the $\pm 3\sigma$ bound where σ is the standard deviation computed using 100 prediction realizations.

Next, a multistart Newton iteration is initiated as per Algorithm 1 to compute the sparse representation of damping coefficients. Figure 12 show the NSBL results for three different choices of starting $\log \boldsymbol{\alpha}$ value. A unique solution of $\boldsymbol{\alpha}^{\text{map}} = \{-5.02, 4.03, 4.42\}$ is obtained from multistart Newton iterations. At this optimum, the relevance indicator for the damping coefficients c_2 and c_3 converges to a value of zero, indicating their irrelevance towards the system physics. The relevance indicator for c_1 approaches the value of one, indicating its relevance. Also note that the log-evidence and objective function values are identical for the optimum, demonstrating that the sparse learning process is solely dictated by model evidence. In summary, the sparse relevant parameter vector is identified as $\boldsymbol{\phi} = \{c_1, k_1, k_2, k_3\}$, which is identical to the data-generating model. Notice that this sparse structure was identified while accounting for the non-Gaussian behavior of posterior pdf and the prior knowledge regarding stiffness coefficients. This aspect distinguishes NSBL from SBL and BCS, which are incapable of handling non-Gaussian parameter pdfs and prior knowledge.

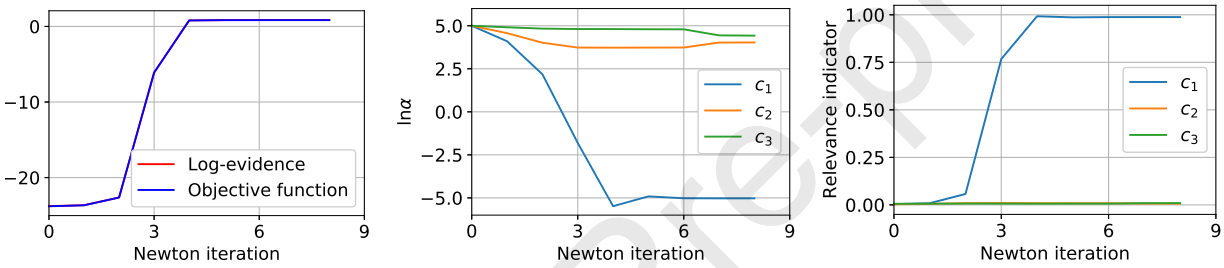
As derived in Section 2.4.2, the posterior pdf of the relevant parameters is also available analytically as a sum of 2500 Gaussian kernels with an updated mean and covariance. This posterior pdf is computed using the user-supplied prior pdf for stiffness coefficients k_i (listed in Table 4), and a data-informed ARD prior $\mathcal{N}(c_1|0, e^{5.02})$ using the optimal hyperparameter value. Figure 13 shows the marginal pdf of the relevant parameters. Figure 14 shows the predicted time-history of velocity $\dot{u}_1$. Notice the reduction in variance in predictions before (Figure 11) and after (Figure 14) sparse learning. This reduction in variance is the result of removal of redundant parameters c_2 and c_3 that were causing overfitting.



(a) $\boldsymbol{\alpha}$ initiated at $\{-5, -5, -5\}$



(b) $\boldsymbol{\alpha}$ initiated at $\{0, 0, 0\}$



(c) $\boldsymbol{\alpha}$ initiated at $\{5, 5, 5\}$

Figure 12: NSBL results for identifying sparsity among damping coefficients of the three-dof mass-spring-damper model. Notice that log-evidence and objective function values are identical for the range of $\log \boldsymbol{\alpha}$ values attempted during Newton iteration; hence the plots are indistinguishable.

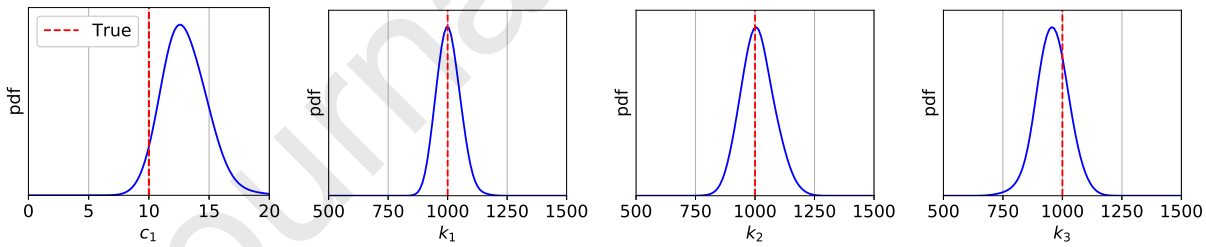


Figure 13: Marginal posterior pdf for the relevant parameters of the three-dof mass-spring-damper model.

4. Conclusion

Nonlinear sparse Bayesian learning (NSBL) is offered as a computationally efficient alternative to sampling-based sparse learning of physics-based models. NSBL can also be considered as an extension of SBL to nonlinear models. The analytical tractability of the Bayesian analysis is enabled by a GMM approximation of the partial posterior pdf $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi})\mathrm{p}(\boldsymbol{\phi}_{\text{-}\alpha})$, and the subsequent assignment of Gaussian ARD priors to question-

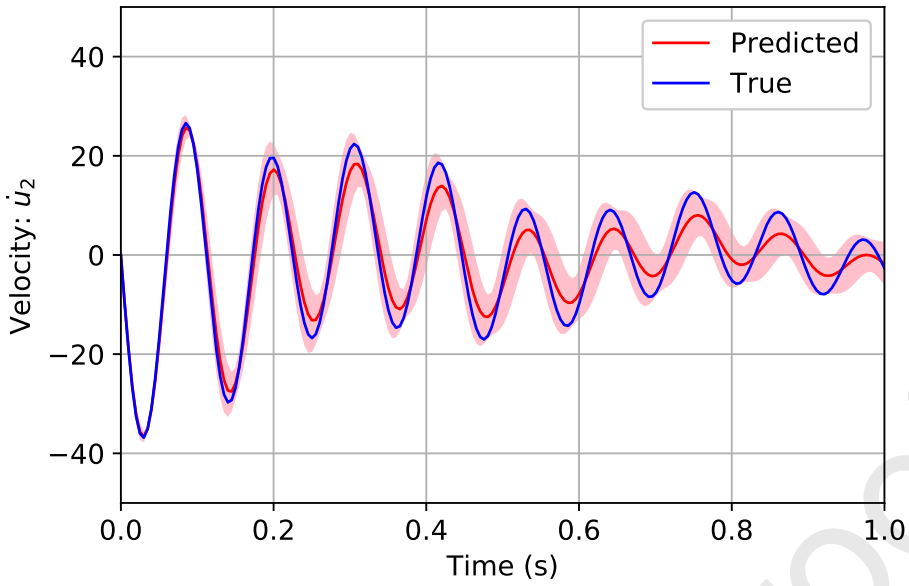


Figure 14: Predicted versus true time-history of velocity at first storey using the optimal nested model.

able parameters. A detailed mathematical derivation into the semi-analytical solution for model evidence, posterior parameter pdf, and the gradient and Hessian information of model evidence was presented in this paper. SBL and NSBL were shown (numerically and analytically) to be the same for a linear regression setting. NSBL was validated using a polynomial regression example involving multimodal posterior pdf and a multimodal model evidence. This numerical investigation demonstrated the sparse learning capabilities of NSBL in physics-based applications involving non-Gaussian posterior and prior pdfs. Finally, the NSBL algorithm was applied to identify a sparse representation of damping coefficients for a three-dof mass-spring-damper model of a shear building frame. The true sparse structure of damping coefficients was identified by NSBL while the posterior pdf for stiffness coefficients were found to be centered around the true values. In summary, NSBL provides an efficient alternative to alleviating overfitting during Bayesian inversion of complex physical systems.

5. Acknowledgement

The first author acknowledges the support of Ontario Graduate Scholarship program and the Canadian Department of National Defence. The fourth author acknowledges the support of the Canadian Department of National Defence and a Discovery Grant from Natural Sciences and Engineering Research Council of Canada. The fifth author acknowledges the support of a Discovery Grant from Natural Sciences and Engineering Research Council of Canada. The computing infrastructure is supported by the Canada Foundation for Innovation (CFI), the Ontario Innovation Trust (OIT), CLUMEQ and SciNet HPC Consortia at Canada.

Appendix A. Summary of SBL and BCS algorithms

Consider that a linear regression model is given as $\mathbf{d} = \boldsymbol{\Psi}\boldsymbol{\phi} + \boldsymbol{\epsilon}$, where $\boldsymbol{\phi} \in \mathbb{R}^{N_\phi}$ is the unknown coefficient vector, $\mathcal{D} \equiv \mathbf{d} \in \mathbb{R}^{N_d \times 1}$ is the measurement vector, $\boldsymbol{\Psi} \in \mathbb{R}^{N_d \times N_\phi}$ is the design matrix, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \rho^{-1}\mathbf{I}_{N_d})$ is the model error where ρ is the error precision. Likelihood function is then known as $\mathrm{p}(\mathcal{D}|\boldsymbol{\phi}) = \mathcal{N}(\mathbf{d}|\boldsymbol{\Psi}\boldsymbol{\phi}, \rho^{-1}\mathbf{I}_{N_d})$. The hierarchical prior assignment for SBL and BCS is summarized in Table A.1. Note that $\beta_i = 1/\alpha_i$ in Table A.1. Following this prior definition, the posterior pdf can be computed using Bayesian linear regression as $\mathrm{p}(\boldsymbol{\phi}|\mathbf{d}, \boldsymbol{\alpha}, \rho) = \mathcal{N}(\boldsymbol{\phi}|\mathbf{m}, \mathbf{P})$ where $\mathbf{P} = (\mathbf{I}_{N_\phi} - \mathbf{K}\Phi)\mathbf{A}^{-1}$,

$\mathbf{m} = \mathbf{K}\mathbf{y}$, $\mathbf{K} = \mathbf{A}^{-1}\Phi^T\mathbf{B}^{-1}$ and $\mathbf{B} = \Phi\mathbf{A}^{-1}\Phi^T + \mathbf{I}_{N_d}\rho^{-1}$. The model evidence is also available analytically as $\mathrm{p}(\mathbf{y}|\boldsymbol{\alpha}, \rho) = \mathcal{N}(\mathbf{y}|\mathbf{0}, \mathbf{B})$. Both SBL and BCS algorithms operate by updating α_i or β_i at each iteration, where the choice for an appropriate index i can be made randomly or deterministicly. Each update in α_i is followed by update in parameter posterior statistics ($\mathbf{m}$ and $\mathbf{P}$), error precision ρ, and the hyperprior parameter λ (only for BCS). As evident from the cost functions listed in Table A.1, SBL is a special case of BCS when λ is fixed at zero [16]. Also, the sparsity inducing marginal prior pdf is different for SBL and BCS. BCS uses Laplace distribution while SBL uses Student's-t distribution [14, 15]. From Table A.1, the cost function for BCS has an additional term of $-\lambda\beta_i$ in comparison to SBL. This additional term entails additional penalty on large β_i values, thereby producing sparser solutions than those obtained for SBL [16].

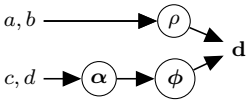
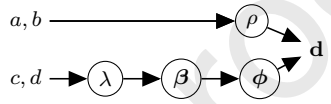
Entity	SBL/RVM	BCS
Hierarchical prior pdf	$\mathrm{p}(\boldsymbol{\phi}\|\boldsymbol{\alpha}) = \mathcal{N}(\boldsymbol{\phi}\|\mathbf{0}, \mathbf{A}^{-1})$ $\mathbf{A} = \mathrm{Diag}(\boldsymbol{\alpha})$ $\mathrm{p}(\alpha_i) = \mathcal{G}(\alpha_i\|r_i, s_i)$ $\mathrm{p}(\rho) = \mathcal{G}(\rho\|a, b)$ 	$\mathrm{p}(\boldsymbol{\phi}\|\boldsymbol{\beta}) = \mathcal{N}(\boldsymbol{\phi}\|\mathbf{0}, \mathbf{A}^{-1})$ $\mathbf{A}^{-1} = \mathrm{Diag}(\boldsymbol{\beta})$ $\mathrm{p}(\beta_i\|\lambda) = \mathcal{G}(\beta_i\|1, \lambda/2)$ $\mathrm{p}(\lambda) = \mathcal{G}(\lambda\|c, d)$ $\mathrm{p}(\rho) = \mathcal{G}(\rho\|a, b)$ 
Marginal prior pdf of ϕ_i	Student's-t $\mathrm{p}(w_i) = \mathcal{T}(w_i\|0, 2c, d/c)$	Laplace $\mathrm{p}(w_i\|\lambda) = \mathcal{LP}(w_i\|0, \lambda^{-0.5})$
Cost function (in terms of β_i)	$\frac{1}{2}\left(\log\left(\frac{1}{1+\beta_i s_i}\right) + \frac{q_i^2\beta_i}{1+\beta_i s_i}\right)$	$\frac{1}{2}\left(\log\left(\frac{1}{1+\beta_i s_i}\right) + \frac{q_i^2\beta_i}{1+\beta_i s_i} - \lambda\beta_i\right)$
Implementation	fastSBL [23]	fastLAPLACE [16]

Table A.1: Comparison of SBL/RVM and BCS algorithms. $\mathcal{LP}(x|r, s)$ is a laplace distribution with pdf $\exp(-|x-r|/s)/2s$, where s is the scale parameter. $\mathcal{T}(x|\mu, \nu, V)$ is a student's t-distribution with mean μ, dof ν and shape parameter V (section 7.7 in [36]).

Appendix B. Analytical calculations

Appendix B.1. Model evidence

Using the parameter decomposition $\boldsymbol{\phi} = \{\boldsymbol{\phi}_\alpha, \boldsymbol{\phi}_{\text{-}\alpha}\}$, the Gaussian kernel $\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ in Eq. (8) is rewritten in an expanded form as

$$\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}) = \mathcal{N}\left(\begin{Bmatrix}\boldsymbol{\phi}_\alpha \\ \boldsymbol{\phi}_{\text{-}\alpha}\end{Bmatrix} \middle| \begin{Bmatrix}\boldsymbol{\mu}_\alpha^{(k)} \\ \boldsymbol{\mu}_{\text{-}\alpha}^{(k)}\end{Bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_\alpha^{(k)} & \mathbf{C}^{(k)} \\ (\mathbf{C}^{(k)})^T & \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)}\end{bmatrix}\right), \tag{B.1}$$

where $\boldsymbol{\mu}_\alpha^{(k)} \in \mathbb{R}^{N_\alpha}$ and $\boldsymbol{\Sigma}_\alpha^{(k)} \in \mathbb{R}^{N_\alpha \times N_\alpha}$ pertain to the questionable parameters $\boldsymbol{\phi}_\alpha$; $\boldsymbol{\mu}_{\text{-}\alpha}^{(k)} \in \mathbb{R}^{N_\phi - N_\alpha}$ and $\boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} \in \mathbb{R}^{(N_\phi - N_\alpha)\times(N_\phi - N_\alpha)}$ pertain to the *a priori* relevant parameters $\boldsymbol{\phi}_{\text{-}\alpha}$; and $\mathbf{C}^{(k)} \in \mathbb{R}^{N_\alpha \times (N_\phi - N_\alpha)}$ is the cross-covariance matrix of $\boldsymbol{\phi}_\alpha$ and $\boldsymbol{\phi}_{\text{-}\alpha}$. Using the conditional distribution relations for multivariate Gaussian pdfs, the Gaussian kernel in Eq. (B.1) is re-written as (using $\mathrm{p}(X_1, X_2) = \mathrm{p}(X_1|X_2)\mathrm{p}(X_2)$)

$$\mathcal{N}(\boldsymbol{\phi}|\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}) = \mathcal{N}(\boldsymbol{\phi}_{\text{-}\alpha}|\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}, \tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)})\mathcal{N}(\boldsymbol{\phi}_\alpha|\boldsymbol{\mu}_\alpha^{(k)}, \boldsymbol{\Sigma}_\alpha^{(k)}), \tag{B.2a}$$

$$\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)} = \boldsymbol{\mu}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}(\boldsymbol{\phi}_\alpha - \boldsymbol{\mu}_\alpha^{(k)}), \tag{B.2b}$$

$$\tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)} = \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} - (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}\mathbf{C}^{(k)}, \tag{B.2c}$$

where the mean vector $\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}$ and the covariance matrix $\tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}$ pertain to $\boldsymbol{\phi}_{\text{-}\alpha}$ conditioned on a known $\boldsymbol{\phi}_\alpha$ value. Matrix $\tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}$ is also known as the schur complement of matrix $\boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)}$ defined in Eq. (B.1).

Substituting the expansion from Eq. (B.2a) in Eq. (8) leads to

$$\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha}) = \sum_{k=1}^{K} a^{(k)} \int \mathcal{N}(\boldsymbol{\phi}_{\text{-}\alpha}|\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}, \tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}) \underbrace{\mathcal{N}(\boldsymbol{\phi}_\alpha|\boldsymbol{\mu}_\alpha^{(k)}, \boldsymbol{\Sigma}_\alpha^{(k)})\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1})}_{\text{Product of two Gaussian pdfs}} d\boldsymbol{\phi}. \tag{B.3}$$

The product of two Gaussian pdfs in Eq. (B.3) can be evaluated analytically as [36]

$$\mathcal{N}(\boldsymbol{\phi}_\alpha|\boldsymbol{\mu}_\alpha^{(k)}, \boldsymbol{\Sigma}_\alpha^{(k)})\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{0}, \mathbf{A}^{-1}) = \mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)})\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{m}_\alpha^{(k)}, \mathbf{P}_\alpha^{(k)}), \tag{B.4a}$$

$$\mathbf{B}_\alpha^{(k)} = \boldsymbol{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1}, \tag{B.4b}$$

$$\mathbf{P}_\alpha^{(k)} = \left((\boldsymbol{\Sigma}_\alpha^{(k)})^{-1} + \mathbf{A}\right)^{-1}, \tag{B.4c}$$

$$\mathbf{m}_\alpha^{(k)} = \mathbf{P}_\alpha^{(k)}(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)}, \tag{B.4d}$$

where $\mathbf{m}_\alpha^{(k)}$ is the posterior mean and $\mathbf{P}_\alpha^{(k)}$ is the posterior covariance of questionable parameters $\boldsymbol{\phi}_\alpha$, pertaining to the k^{th} kernel. Notice that $\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)})$ is independent of $\boldsymbol{\phi}$ and acts as a normalizing factor in Eq. (B.4a). Substituting Eq. (B.4a) in Eq. (B.3) and taking $\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)})$ out of the integral leads to

$$\begin{aligned}
\hat{\mathrm{p}}(\mathcal{D}|\boldsymbol{\alpha}) &= \sum_{k=1}^{K} a^{(k)} \int \mathcal{N}(\boldsymbol{\phi}_{\text{-}\alpha}|\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}, \tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)})\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)})\mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{m}_\alpha^{(k)}, \mathbf{P}_\alpha^{(k)})d\boldsymbol{\phi} \\
&= \sum_{k=1}^{K} a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)}) \int \left\{ \int \mathcal{N}(\boldsymbol{\phi}_{\text{-}\alpha}|\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}, \tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)})d\boldsymbol{\phi}_{\text{-}\alpha} \right\}}^{1} \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{m}_\alpha^{(k)}, \mathbf{P}_\alpha^{(k)})d\boldsymbol{\phi}_\alpha \\
&= \sum_{k=1}^{K} a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)}) \int \mathcal{N}(\boldsymbol{\phi}_\alpha|\mathbf{m}_\alpha^{(k)}, \mathbf{P}_\alpha^{(k)})d\boldsymbol{\phi}_\alpha}^{1} \\
&= \sum_{k=1}^{K} a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0}, \mathbf{B}_\alpha^{(k)})
\end{aligned} \tag{B.5}$$

where $a^{(k)}$ is known from Eq. (3), $\boldsymbol{\mu}_\alpha^{(k)}$ is known from Eq. (B.1), and $\mathbf{B}_\alpha^{(k)}$ is known from Eq. (B.4b).

Appendix B.2. Posterior parameter pdf

Using the Woodbury identity (Eq. 156 in [36]), the posterior covariance $\mathbf{P}_\alpha^{(k)}$ of $\boldsymbol{\phi}_\alpha$ in Eq. (B.4c) can be rewritten as

$$\mathbf{P}_\alpha^{(k)} = \boldsymbol{\Sigma}_\alpha^{(k)} - \boldsymbol{\Sigma}_\alpha^{(k)}(\mathbf{B}_\alpha^{(k)})^{-1}\boldsymbol{\Sigma}_\alpha^{(k)} \tag{B.6}$$

Using this expansion, the posterior mean $\mathbf{m}_\alpha^{(k)}$ of $\boldsymbol{\phi}_\alpha$ in Eq. (B.4d) can be rewritten as

$$\begin{aligned}
\mathbf{m}_\alpha^{(k)} = \mathbf{P}_\alpha^{(k)}(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)} &= \left(\boldsymbol{\Sigma}_\alpha^{(k)} - \boldsymbol{\Sigma}_\alpha^{(k)}(\mathbf{B}_\alpha^{(k)})^{-1}\boldsymbol{\Sigma}_\alpha^{(k)}\right)(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)} \\
&= \boldsymbol{\mu}_\alpha^{(k)} - \boldsymbol{\Sigma}_\alpha^{(k)}(\mathbf{B}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)},
\end{aligned} \tag{B.7}$$

where $\mathbf{B}_\alpha^{(k)} = \boldsymbol{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1}$ is known from Eq. (B.4b).

Next, using the law of total expectation [24], the posterior mean $\mathbf{m}_\alpha^{(k)}$ of $\boldsymbol{\phi}_{\text{-}\alpha}$ is computed by taking the expectation of conditional mean $\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}$ in Eq. (B.2b) as

$$\mathbf{m}_{\text{-}\alpha}^{(k)} = \mathrm{E}_k\left[\mathrm{E}_k\left[\boldsymbol{\phi}_{\text{-}\alpha}|\boldsymbol{\phi}_\alpha\right]\right] = \mathrm{E}_k\left[\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}\right] = \mathrm{E}_k\left[\boldsymbol{\mu}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}(\boldsymbol{\phi}_\alpha - \boldsymbol{\mu}_\alpha^{(k)})\right] \tag{B.8a}$$

$$= \boldsymbol{\mu}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_\alpha^{(k)})^{-1}(\mathbf{m}_\alpha^{(k)} - \boldsymbol{\mu}_\alpha^{(k)}), \tag{B.8b}$$

where the inner expectation is with respect to $\boldsymbol{\phi}_{\text{-}\alpha}$ and the outer expectation is with respect to $\boldsymbol{\phi}_{\alpha}$. Substituting $\mathbf{m}_{\alpha}^{(k)}$ from Eq. (B.7) in Eq. (B.8b) leads to

$$\begin{aligned}\mathbf{m}_{\text{-}\alpha}^{(k)} &= \boldsymbol{\mu}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}(\cancel{\boldsymbol{\mu}_{\alpha}^{(k)}} - \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)} - \cancel{\boldsymbol{\mu}_{\alpha}^{(k)}}) \\ &= \boldsymbol{\mu}_{\text{-}\alpha}^{(k)} - (\mathbf{C}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)} \end{aligned} \tag{B.9}$$

Rewriting Eq. (B.7) and Eq. (B.9) in a matrix form produces the solution to posterior mean vector $\mathbf{m}^{(k)}$ as

$$\mathbf{m}^{(k)} = \begin{Bmatrix}\mathbf{m}_{\alpha}^{(k)} \\ \mathbf{m}_{\text{-}\alpha}^{(k)}\end{Bmatrix} = \begin{Bmatrix}\boldsymbol{\mu}_{\alpha}^{(k)} \\ \boldsymbol{\mu}_{\text{-}\alpha}^{(k)}\end{Bmatrix} - \begin{bmatrix}\boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1} & \mathbf{0} \\ (\mathbf{C}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1} & \mathbf{0}\end{bmatrix}\begin{Bmatrix}\boldsymbol{\mu}_{\alpha}^{(k)} \\ \boldsymbol{\mu}_{\text{-}\alpha}^{(k)}\end{Bmatrix}. \tag{B.10}$$

Next, the posterior covariance $\mathbf{P}_{\text{-}\alpha}^{(k)}$ of $\boldsymbol{\phi}_{\text{-}\alpha}$ in Eq. (12) is computed using the law of total variance [24] as

$$\begin{aligned}\mathbf{P}_{\text{-}\alpha}^{(k)} &= \text{Covar}_k(\boldsymbol{\phi}_{\text{-}\alpha}) = \text{Covar}_k(\text{E}_k\left[\boldsymbol{\phi}_{\text{-}\alpha}|\boldsymbol{\phi}_{\alpha}\right]) + \text{E}_k\left[\text{Covar}_k(\boldsymbol{\phi}_{\text{-}\alpha}|\boldsymbol{\phi}_{\alpha})\right] = \text{Covar}_k(\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}) + \text{E}_k[\tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}] \\ &= \text{Covar}_k\left(\boldsymbol{\mu}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}(\boldsymbol{\phi}_{\alpha} - \boldsymbol{\mu}_{\alpha}^{(k)})\right) + \text{E}_k\left[\boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} - (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)}\right] \\ &= (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}\text{Covar}_k\left(\boldsymbol{\phi}_{\alpha}\right)(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} + \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} - (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} \\ &= \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}\left(\mathbf{P}_{\alpha}^{(k)}(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1} - \mathbf{I}_{N_\alpha}\right)\mathbf{C}^{(k)}, \end{aligned} \tag{B.11}$$

where $\tilde{\boldsymbol{\mu}}_{\text{-}\alpha}^{(k)}$ and $\tilde{\boldsymbol{\Sigma}}_{\text{-}\alpha}^{(k)}$ are substituted from Eq. (B.2). Substituting $\mathbf{P}_{\alpha}^{(k)}$ from Eq. (B.6) in Eq. (B.11) results in

$$\mathbf{P}_{\text{-}\alpha}^{(k)} = \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} + (\mathbf{C}^{(k)})^T(\boldsymbol{\Sigma}_{\alpha}^{(k)})^{-1}\left(-\boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\right)\mathbf{C}^{(k)} = \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)} - (\mathbf{C}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} \tag{B.12}$$

Next, the posterior cross-covariance matrix $\mathbf{D}^{(k)}$ in Eq. (12) is evaluated as

$$\begin{aligned}\mathbf{D}^{(k)} &= \text{E}_k\left[\left(\boldsymbol{\phi}_{\alpha} - \mathbf{m}_{\alpha}^{(k)}\right)\left(\boldsymbol{\phi}_{\text{-}\alpha} - \mathbf{m}_{\text{-}\alpha}^{(k)}\right)^T\right] \\ &= \text{E}_k\left[\left(\boldsymbol{\phi}_{\alpha} - \boldsymbol{\mu}_{\alpha}^{(k)} + \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}\right)\left(\boldsymbol{\phi}_{\text{-}\alpha} - \boldsymbol{\mu}_{\text{-}\alpha}^{(k)} + \mathbf{C}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}\right)^T\right] \\ &= \mathbf{C}^{(k)} + \left(\mathbf{m}_{\alpha}^{(k)} - \boldsymbol{\mu}_{\alpha}^{(k)}\right)(\boldsymbol{\mu}_{\alpha}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} + \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}\left(\mathbf{m}_{\text{-}\alpha}^{(k)} - \boldsymbol{\mu}_{\text{-}\alpha}^{(k)}\right)^T \\ &\quad + \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}(\boldsymbol{\mu}_{\alpha}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)}. \end{aligned} \tag{B.13}$$

Substituting $\mathbf{m}_{\alpha}^{(k)}$ from Eq. (B.7) and $\mathbf{m}_{\text{-}\alpha}^{(k)}$ from Eq. (B.9) reduces Eq. (B.13) to

$$\begin{aligned}\mathbf{D}^{(k)} &= \mathbf{C}^{(k)} - \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}(\boldsymbol{\mu}_{\alpha}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} - \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}(\boldsymbol{\mu}_{\alpha}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} \\ &\quad + \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}(\boldsymbol{\mu}_{\alpha}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\mathbf{C}^{(k)} \\ &= \mathbf{C}^{(k)} - \boldsymbol{\Sigma}_{\alpha}^{(k)}(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}\left((\mathbf{C}^{(k)})^T(\mathbf{B}_{\alpha}^{(k)})^{-1}\boldsymbol{\mu}_{\alpha}^{(k)}\right)^T \end{aligned} \tag{B.14}$$

This completes the calculation of posterior pdf $\hat{\text{p}}(\boldsymbol{\phi}|\mathcal{D}, \boldsymbol{\alpha})$ in Eq. (12), wherein $\mathbf{m}^{(k)}$ is known from Eq. (B.10), $\mathbf{P}_{\alpha}^{(k)}$ is known from Eq. (B.6), $\mathbf{P}_{\text{-}\alpha}^{(k)}$ is known from Eq. (B.11), and $\mathbf{D}^{(k)}$ is known from Eq. (B.14). Notice that when using flat prior for questionable parameters (i.e. prior precision $\boldsymbol{\alpha} \approx \mathbf{0}$), we get $\mathbf{m}_{\alpha}^{(k)} \approx \boldsymbol{\mu}_{\alpha}^{(k)}$, $\mathbf{m}_{\text{-}\alpha}^{(k)} \approx \boldsymbol{\mu}_{\text{-}\alpha}^{(k)}$, $\mathbf{P}_{\alpha}^{(k)} \approx \boldsymbol{\Sigma}_{\alpha}^{(k)}$, $\mathbf{P}_{\text{-}\alpha}^{(k)} \approx \boldsymbol{\Sigma}_{\text{-}\alpha}^{(k)}$ and $\mathbf{D}^{(k)} \approx \mathbf{C}^{(k)}$.

Appendix B.3. Gradient vector

$v_i^{(k)}$ in Eq. (17) is evaluated as

$$v_i^{(k)} = \frac{\partial \log \mathcal{N}(\boldsymbol{\mu}_{\alpha}^{(k)}|\mathbf{0}, \mathbf{B}_{\alpha}^{(k)})}{\partial \log \alpha_i} = -\frac{1}{2}\left\{\frac{\partial \log|\mathbf{B}_{\alpha}^{(k)}|}{\partial \log \alpha_i} + (\boldsymbol{\mu}_{\alpha}^{(k)})^T\frac{\partial(\mathbf{B}_{\alpha}^{(k)})^{-1}}{\partial \log \alpha_i}\boldsymbol{\mu}_{\alpha}^{(k)}\right\}. \tag{B.15}$$

An identity involving the derivative of a matrix determinant is available as (Eq. (46) in [36])

$$\frac{\partial \log |\mathbf{Z}|}{\partial x} = \text{Trace}\left(\mathbf{Z}^{-1}\frac{\partial \mathbf{Z}}{\partial x}\right), \tag{B.16}$$

where $\mathbf{Z}$ is any square matrix and x is a scalar. The derivative of term $\log|\mathbf{B}_\alpha^{(k)}|$ in Eq. (B.15) is evaluated using this identity as

$$\frac{\partial \log |\mathbf{B}_\alpha^{(k)}|}{\partial \log \alpha_i} = \alpha_i \frac{\partial \log |\mathbf{B}_\alpha^{(k)}|}{\partial \alpha_i} = \alpha_i \text{Trace}\left((\mathbf{B}_\alpha^{(k)})^{-1}\frac{\partial(\mathbf{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1})}{\partial \alpha_i}\right), \tag{B.17}$$

where $\mathbf{B}_\alpha^{(k)} = \mathbf{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1}$ is substituted from Eq. (B.4b). As observed from Eq. (3), the matrix $\mathbf{\Sigma}_\alpha^{(k)}$ is solely dictated by the observations $\mathcal{D}$ and the known prior $\text{p}(\boldsymbol{\phi}_{\text{-}\alpha})$, and is not a function of α_i. On the other hand, matrix $\mathbf{A} = \text{Diag}(\boldsymbol{\alpha})$ is a diagonal matrix with $\boldsymbol{\alpha}$ as its diagonal. Hence, differentiating $\mathbf{A}^{-1}$=Diag$(1/\boldsymbol{\alpha})$ with respect to α_i will produce $(-1/\alpha_i^2)\mathbf{\Delta}_{ii}$, where the matrix $\mathbf{\Delta}_{ii} \in \mathbb{R}^{N_\alpha \times N_\alpha}$ has only (i,i) element equal to one and the rest of the elements are all zeros. Consequently, Eq. (B.17) can be evaluated as

$$\frac{\partial \log |\mathbf{B}_\alpha^{(k)}|}{\partial \log \alpha_i} = \alpha_i \text{Trace}\left((\mathbf{B}_\alpha^{(k)})^{-1}\frac{\partial \mathbf{A}^{-1}}{\partial \alpha_i}\right) = -\frac{1}{\alpha_i}\text{Trace}\left((\mathbf{B}_\alpha^{(k)})^{-1}\Delta_{ii}\right). \tag{B.18}$$

Further, the matrix $(\mathbf{B}_\alpha^{(k)})^{-1}$ from Eq. (B.4d) is expanded using the Woodbury identity (Eq. 156 in [36]) to produce

$$(\mathbf{B}_\alpha^{(k)})^{-1} = (\mathbf{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1})^{-1} = \mathbf{A} - \mathbf{A}\mathbf{P}_\alpha^{(k)}\mathbf{A}, \tag{B.19}$$

where $\mathbf{P}_\alpha^{(k)}$ is known from Eq. (B.4b). Substituting Eq. (B.19) in Eq. (B.18) leads to

$$\begin{aligned}\frac{\partial \log |\mathbf{B}_\alpha^{(k)}|}{\partial \log \alpha_i} &= -\frac{1}{\alpha_i}\text{Trace}\left(\mathbf{A}\mathbf{\Delta}_{ii} - \mathbf{A}\mathbf{P}_\alpha^{(k)}\mathbf{A}\mathbf{\Delta}_{ii}\right) \\ &= -\frac{1}{\alpha_i}\left(\alpha_i - \alpha_i^2 P_{ii}^{(k)}\right) = -1 + \alpha_i P_{ii}^{(k)},\end{aligned} \tag{B.20}$$

where the variance $P_{ii}^{(k)}$ is the (i,i) element of matrix $\mathbf{P}_\alpha^{(k)}$ or $\mathbf{P}^{(k)}$ (see Eq. (12)).

Next, the derivative of $(\mathbf{B}_\alpha^{(k)})^{-1}$ in Eq. (B.15) is evaluated by expanding $(\mathbf{B}_\alpha^{(k)})^{-1}$ using the Woodbury identity [36] as

$$(\mathbf{B}_\alpha^{(k)})^{-1} = (\mathbf{\Sigma}_\alpha^{(k)} + \mathbf{A}^{-1})^{-1} = (\mathbf{\Sigma}_\alpha^{(k)})^{-1} - (\mathbf{\Sigma}_\alpha^{(k)})^{-1}\mathbf{P}_\alpha^{(k)}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}, \tag{B.21}$$

where only the posterior covariance $\mathbf{P}_\alpha^{(k)}$ depends on α_i. As a result,

$$\begin{aligned}(\boldsymbol{\mu}_\alpha^{(k)})^T\frac{\partial(\mathbf{B}_\alpha^{(k)})^{-1}}{\partial \log \alpha_i}\boldsymbol{\mu}_\alpha^{(k)} &= \alpha_i(\boldsymbol{\mu}_\alpha^{(k)})^T\left\{\frac{\partial}{\partial \alpha_i}\left((\mathbf{\Sigma}_\alpha^{(k)})^{-1} - (\mathbf{\Sigma}_\alpha^{(k)})^{-1}\mathbf{P}^{(k)}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\right)\right\}\boldsymbol{\mu}_\alpha^{(k)} \\ &= -\alpha_i(\boldsymbol{\mu}_\alpha^{(k)})^T(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\frac{\partial \mathbf{P}^{(k)}}{\partial \alpha_i}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)}.\end{aligned} \tag{B.22}$$

The derivative of $\mathbf{P}_\alpha^{(k)}$ is evaluated by using an identity involving the derivative of matrix inverse: (Eq. (59) in [36]),

$$\frac{\partial \mathbf{Z}^{-1}}{\partial x} = -\mathbf{Z}^{-1}\frac{\partial \mathbf{Z}}{\partial x}\mathbf{Z}^{-1}, \tag{B.23}$$

where $\mathbf{Z}$ is any square matrix and x is a scalar. Using this identity in Eq. (B.22) leads to

$$\begin{aligned}(\boldsymbol{\mu}_\alpha^{(k)})^T\frac{\partial(\mathbf{B}_\alpha^{(k)})^{-1}}{\partial \log \alpha_i}\boldsymbol{\mu}_\alpha^{(k)} &= -\alpha_i(\boldsymbol{\mu}_\alpha^{(k)})^T(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\left\{-\mathbf{P}_\alpha^{(k)}\frac{\partial(\mathbf{P}_\alpha^{(k)})^{-1}}{\partial \alpha_i}\mathbf{P}_\alpha^{(k)}\right\}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)} \\ &= \alpha_i\left\{\mathbf{P}_\alpha^{(k)}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)}\right\}^T\left\{\frac{\partial}{\partial \alpha_i}\left((\mathbf{\Sigma}_\alpha^{(k)})^{-1} + \mathbf{A}\right)\right\}\left\{\mathbf{P}_\alpha^{(k)}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)}\right\} \\ &= \alpha_i(\mathbf{m}_\alpha^{(k)})^T\mathbf{\Delta}_{ii}\mathbf{m}_\alpha^{(k)} = \alpha_i(m_i^{(k)})^2,\end{aligned} \tag{B.24}$$

where $(\mathbf{P}_\alpha^{(k)})^{-1} = (\mathbf{\Sigma}_\alpha^{(k)})^{-1} + \mathbf{A}$ is substituted from Eq. (B.4b), $\mathbf{m}_\alpha^{(k)} = \mathbf{P}_\alpha^{(k)}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)}$ is substituted from Eq. (B.4d), and $m_i^{(k)}$ is the i^{th} element of posterior mean vector $\mathbf{m}_\alpha^{(k)}$. Substituting Eq. (B.20) and Eq. (B.24) in Eq. (B.15) leads to

$$v_i^{(k)} = -\frac{1}{2}\left\{-1 + \alpha_i P_{ii}^{(k)} + \alpha_i (m_i^{(k)})^2\right\} = \frac{\gamma_i^{(k)} - \alpha_i (m_i^{(k)})^2}{2}\,, \tag{B.25}$$

where $\gamma_i^{(k)} = 1 - \alpha_i P_{ii}^{(k)}$ is defined as the relevance indicator for parameter $\boldsymbol{\phi}_i$ corresponding to the k^{th} kernel.

Appendix B.4. Hessian matrix

The derivative of the factor $v_j^{(k)}$ in Eq. (20) is obtained using Eq. (18) as

$$\frac{\partial v_j^{(k)}}{\partial \log \alpha_i} = \frac{\alpha_i}{2}\left\{-\alpha_j\left(\frac{\partial P_{jj}^{(k)}}{\partial \alpha_i} + \frac{\partial (m_j^{(k)})^2}{\partial \alpha_i}\right) - \delta_{ij}\left(P_{ii}^{(k)} + (m_i^{(k)})^2\right)\right\}\,. \tag{B.26}$$

Note that $\mathbf{\Delta}_{ii}$ matrix can also be written as $\boldsymbol{\delta}_i\boldsymbol{\delta}_i^T$ where $\boldsymbol{\delta}_i \in \mathbb{R}^{N_\alpha}$ has the i^{th} element as one and rest are all zeros. Using this definition, the derivative of the posterior variance $P_{jj}^{(k)}$ in Eq. (B.26) is computed as

$$\begin{aligned}\frac{\partial P_{jj}^{(k)}}{\partial \alpha_i} &= \frac{\partial}{\partial \alpha_i}\left(\boldsymbol{\delta}_j^T \mathbf{P}_\alpha^{(k)} \boldsymbol{\delta}_j\right) = \boldsymbol{\delta}_j^T\left(\frac{\partial}{\partial \alpha_i}\mathbf{P}_\alpha^{(k)}\right)\boldsymbol{\delta}_j = \boldsymbol{\delta}_j^T\left(-\mathbf{P}_\alpha^{(k)}\mathbf{\Delta}_{ii}\mathbf{P}_\alpha^{(k)}\right)\boldsymbol{\delta}_j \\ &= -\left(\boldsymbol{\delta}_j^T \mathbf{P}_\alpha^{(k)} \boldsymbol{\delta}_i\right)\left(\boldsymbol{\delta}_i^T \mathbf{P}_\alpha^{(k)} \boldsymbol{\delta}_j\right) = -P_{ji}^{(k)} P_{ij}^{(k)} = -(P_{ij}^{(k)})^2\,,\end{aligned} \tag{B.27}$$

where $P_{ij}^{(k)}$ is the (i,j) element of matrix $\mathbf{P}_\alpha^{(k)}$ and $\partial \mathbf{P}_\alpha^{(k)}/\partial \alpha_i = -\mathbf{P}_\alpha^{(k)}\mathbf{\Delta}_{ii}\mathbf{P}_\alpha^{(k)}$ has been previously evaluated in Eq. (B.24). Next, the derivative of $(m_j^{(k)})^2$ in Eq. (B.26) is evaluated using Eq. (B.4c) as

$$\begin{aligned}\frac{\partial (m_j^{(k)})^2}{\partial \alpha_i} &= \frac{\partial}{\partial \alpha_i}\left((\mathbf{m}_\alpha^{(k)})^T \mathbf{\Delta}_{jj} \mathbf{m}_\alpha^{(k)}\right) = 2(\mathbf{m}_\alpha^{(k)})^T \mathbf{\Delta}_{jj}\frac{\partial \mathbf{m}_\alpha^{(k)}}{\partial \alpha_i} \\ &= 2(\mathbf{m}_\alpha^{(k)})^T \mathbf{\Delta}_{jj}\frac{\partial \mathbf{P}_\alpha^{(k)}}{\partial \alpha_i}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)} = 2(\mathbf{m}_\alpha^{(k)})^T \mathbf{\Delta}_{jj}\left(-\mathbf{P}_\alpha^{(k)}\mathbf{\Delta}_{ii}\mathbf{P}_\alpha^{(k)}\right)(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)} \\ &= -2\underbrace{(\mathbf{m}_\alpha^{(k)})^T \boldsymbol{\delta}_j}_{m_j^{(k)}}\underbrace{\boldsymbol{\delta}_j^T \mathbf{P}_\alpha^{(k)} \boldsymbol{\delta}_i}_{P_{ij}^{(k)}}\underbrace{\boldsymbol{\delta}_i^T \mathbf{P}_\alpha^{(k)}(\mathbf{\Sigma}_\alpha^{(k)})^{-1}\boldsymbol{\mu}_\alpha^{(k)}}_{m_i^{(k)}} = -2m_i^{(k)} m_j^{(k)} P_{ij}^{(k)}\end{aligned} \tag{B.28}$$

Eq. (B.26) is then re-written using Eq. (B.27) and Eq. (B.28) as

$$\begin{aligned}\frac{\partial v_j^{(k)}}{\partial \log \alpha_i} &= \frac{\alpha_i}{2}\left\{-\alpha_j\left(-(P_{ij}^{(k)})^2 - 2m_i^{(k)} m_j^{(k)} P_{ij}^{(k)}\right) - \delta_{ij}\left(P_{ii}^{(k)} + (m_i^{(k)})^2\right)\right\} \\ &= \alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)} m_j^{(k)} P_{ij}^{(k)}\right) + \delta_{ij}\left(\frac{-\alpha_i P_{ii}^{(k)} - \alpha_i (m_i^{(k)})^2}{2}\right) \\ &= \alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)} m_j^{(k)} P_{ij}^{(k)}\right) + \delta_{ij}\left(v_i^{(k)} - \frac{1}{2}\right)\end{aligned} \tag{B.29}$$

Next, derivative of weight coefficient $w^{(k)}$ in Eq. (20) is evaluated by expanding $w^{(k)}$ according to Eq. (11) as

$$\begin{aligned}\frac{\partial w^{(k)}}{\partial \log \alpha_i} &= \frac{\partial}{\partial \log \alpha_i}\left\{\frac{a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\right\}\\ &= \frac{a^{(k)}}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\frac{\partial\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\partial\log\alpha_i} - \frac{a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{(\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha}))^2}\frac{\partial\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}{\partial\log\alpha_i}\\ &= \frac{a^{(k)}\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}\left(\frac{\partial\log\mathcal{N}(\boldsymbol{\mu}_\alpha^{(k)}|\mathbf{0},\mathbf{B}_\alpha^{(k)})}{\partial\log\alpha_i} - \frac{\partial\log\hat{\mathrm{p}}(\mathcal{D}|\log\boldsymbol{\alpha})}{\partial\log\alpha_i}\right)\\ &= w^{(k)}\left(v_i^{(k)} - \bar{v}_i\right) \end{aligned} \tag{B.30}$$

where $v_i^{(k)}$ is substitute from Eq. (18), and $\bar{v}_i$ is substituted from Eq. (19b). Using Eq. (B.29) and Eq. (B.30), the Hessian $H_{ij}(\log\boldsymbol{\alpha})$ from Eq. (20) is further evaluated as

$$\begin{aligned}H_{ij}(\log\boldsymbol{\alpha}) = \sum_{k=1}^{K}\Bigg[&w^{(k)}\left\{\alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)}m_j^{(k)}P_{ij}^{(k)}\right) + \delta_{ij}\left(v_i^{(k)} - \frac{1}{2}\right)\right\}\\ &+ v_j^{(k)}\left\{w^{(k)}\left(v_i^{(k)} - \bar{v}_i\right)\right\}\Bigg] - \delta_{ij}s_i\alpha_i\end{aligned} \tag{B.31}$$

Re-arranging terms and substituting $\bar{v}_i$ from Eq. (19) leads to

$$\begin{aligned}H_{ij}(\log\boldsymbol{\alpha}) &= \sum_{k=1}^{K} w^{(k)}\alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)}m_j^{(k)}P_{ij}^{(k)}\right) + \sum_{k=1}^{K} w^{(k)}v_i^{(k)}v_j^{(k)}\\ &\quad - \bar{v}_i\sum_{k=1}^{K} w^{(k)}v_j^{(k)} + \delta_{ij}\left\{\sum_{k=1}^{K} w^{(k)}v_i^{(k)} - \frac{1}{2}\sum_{k=1}^{K} w^{(k)} - s_i\alpha_i\right\}\\ &= \sum_{k=1}^{K} w^{(k)}\left\{\alpha_i\alpha_j\left(\frac{(P_{ij}^{(k)})^2}{2} + m_i^{(k)}m_j^{(k)}P_{ij}^{(k)}\right) + v_i^{(k)}v_j^{(k)} - \bar{v}_i\bar{v}_j\right\}\\ &\quad + \delta_{ij}\left\{\bar{v}_i - \frac{1}{2} - s_i\alpha_i\right\},\end{aligned} \tag{B.32}$$

where $\bar{v}_i$ or $\bar{v}_j$ is known from Eq. (19); $w^{(k)}$ is known from Eq. (11); $v_i^{(k)}$ is known from Eq. (18); $m_i^{(k)}$ or $m_j^{(k)}$ (elements of $\mathbf{m}_\alpha^{(k)}$) is known from Eq. (B.4d); and $P_{ij}^{(k)}$ (elements of $\mathbf{P}_\alpha^{(k)}$) is known from Eq. (B.4c).

Appendix C. PCE surrogate and Sobol sensitivity indices

Given a d-dimensional input $\mathbf{X}$, the PCE surrogate of a scalar output Y is written as

$$Y = \sum_{k=0}^{P-1} y_k\Psi_k(\boldsymbol{\xi}) + \boldsymbol{\epsilon} \tag{C.1}$$

where y_k are the unknown PCE coefficients, $\Psi_k(\boldsymbol{\xi})$ is the k^{th} PCE basis, $\boldsymbol{\xi}$ is the set of d standardized random variables (called germs), P is the total number of PCE basis, and $\boldsymbol{\epsilon}$ is the model discrepancy error. The PCE basis $\Psi(\boldsymbol{\xi})$ are obtained through the tensorization of univariate PCE polynomials chosen from the Askey family of polynomials based on the probability distribution of germs $\boldsymbol{\xi}$ [37]. A total-order PCE truncation with input dimension d and maximum order of p produces $P = (p+d)!/(p!d!)$ number of PCE

basis in Eq. (C.1). The orthogonality of PCE bases with respect to a probability measure helps power many desirable properties needed for performing uncertainty quantification and sensitivity analysis tasks [33, 38]. One such property of PCE surrogates is the analytical availability of Sobol sensitivity indices following the estimation of PCE coefficients [39].

Consider a PCE surrogate with dimension $d = 2$ and order $p = 3$. The standardized input (germ) vector is denoted as $\boldsymbol{\xi} = \{\xi_1, \xi_2\}$. The univariate PCE polynomials of j^{th} input ξ_j are denoted as $\psi_i(\xi_j)$ where i is the polynomial degree. The number of PCE terms for a total-order truncation will be $(2+3)!/(2!3!) = 10$. Each PCE basis $\Psi_k(\boldsymbol{\xi})$ is obtained through tensorization of these univariate PCE polynomials $\psi_i(\xi_j)$ obtained from Askey family of polynomials [38]. The choice of these PCE polynomials is dictated by the probability distribution of germs $\boldsymbol{\xi}$ [33]. The coefficient pertaining to PCE basis $\Psi_k(\boldsymbol{\xi})$ is denoted as y_k. The variance of each PCE basis $\Psi_k(\boldsymbol{\xi})$ is known analytically (product of variance of univariate polynomials) and is denoted as $||\Psi_k||^2$. Table C.1 illustrates the variance decomposition principle of computing Sobol indices through the PCE surrogate for this simple two-dimensional input case.

Order	k	Multi-index $\boldsymbol{\alpha}_k$	PCE basis $\Psi_k(\boldsymbol{\xi})$	Variance contribution solely due to ξ_1	ξ_2	interaction
0	0	[0,0]	1.0			
1	1	[1,0]	$\psi_1(\xi_1)$	$y_1^2\|\|\Psi_1\|\|^2$		
	2	[0,1]	$\psi_1(\xi_2)$		$y_2^2\|\|\Psi_2\|\|^2$	
2	3	[2,0]	$\psi_2(\xi_1)$	$y_3^2\|\|\Psi_3\|\|^2$		
	4	[1,1]	$\psi_1(\xi_1)\psi_1(\xi_2)$			$y_4^2\|\|\Psi_4\|\|^2$
	5	[0,2]	$\psi_2(\xi_2)$		$y_5^2\|\|\Psi_5\|\|^2$	
3	6	[3,0]	$\psi_3(\xi_1)$	$y_6^2\|\|\Psi_6\|\|^2$		
	7	[2,1]	$\psi_2(\xi_1)\psi_1(\xi_2)$			$y_7^2\|\|\Psi_7\|\|^2$
	8	[1,2]	$\psi_1(\xi_1)\psi_2(\xi_2)$			$y_8^2\|\|\Psi_8\|\|^2$
	9	[0,3]	$\psi_3(\xi_2)$		$y_9^2\|\|\Psi_9\|\|^2$	
				V_1	V_2	V_{12}

Table C.1: Sobol indices calculation from PCE surrogates through variance decomposition.

Given the PCE coefficients are available, the Sobol indices are obtained using Table C.1 as [39]

$$\text{Total output variance: } V_{\text{total}} = V_1 + V_2 + V_{12}$$

$$\text{First-order Sobol indices: } S_1 = \frac{V_1}{V_{\text{total}}} \;\;,\;\; S_2 = \frac{V_2}{V_{\text{total}}}$$

$$\text{Second-order Sobol index: } S_{12} = \frac{V_{12}}{V_{total}}$$

$$\text{Total-order Sobol index: } S_1^T = S_1 + S_{12} \;\;,\;\; S_2^T = S_2 + S_{12}$$

References

[1] M. Kennedy, A. O'Hagan, Bayesian calibration of computer models, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63 (2001) 425–464.

[2] K. Yuen, Bayesian Methods for Structural Dynamics and Civil Engineering, Wiley, 2010. URL: https://books.google.ca/books?id=0iDSezuV9mIC.

[3] G. Box, G. Tiao, Bayesian inference in statistical analysis, volume 40, John Wiley & Sons, 2011.

[4] J. Beck, L. Katafygiotis, Updating Models and Their Uncertainties. I: Bayesian Statistical Framework, Journal of Engineering Mechanics 124 (1998) 455–461.

[5] M. Khalil, D. Poirel, A. Sarkar, Probabilistic parameter estimation of a fluttering aeroelastic system in the transitional Reynolds number regime, Journal of Sound and Vibration 332 (2013) 3670–3691.

[6] J. Beck, Bayesian system identification based on probability logic, Structural Control and Health Monitoring 17 (2010) 825–847.

[7] J. Beck, K. Yuen, Model selection using response measurements: Bayesian probabilistic approach, Journal of Engineering Mechanics 41 (2004) 192–203.

[8] M. Khalil, Bayesian Inference for Complex and Large-scale Engineering Systems, Ph.D. thesis, Carleton University, 2013.

[9] S. H. Cheung, J. Beck, Calculation of Posterior Probabilities for Bayesian Model Class Assessment and Averaging from Posterior Samples Based on Dynamic System Data, Computer-Aided Civil and Infrastructure Engineering 25 (2010) 304–321.

[10] K. P. Murphy, Machine learning : a probabilistic perspective, MIT Press, Cambridge, Mass. [u.a.], 2013.

[11] R. Sandhu, D. Poirel, C. Pettit, M. Khalil, A. Sarkar, Bayesian inference of nonlinear unsteady aerodynamics from aeroelastic limit cycle oscillations, Journal of Computational Physics 316 (2016) 534 – 557.

[12] C. C. Liu, M. Aitkin, Bayes factors: Prior sensitivity and model generalizability, Journal of Mathematical Psychology 52 (2008) 362 – 375.

[13] R. Sandhu, C. Pettit, M. Khalil, D. Poirel, A. Sarkar, Bayesian model selection using automatic relevance determination for nonlinear dynamical systems, Computer Methods in Applied Mechanics and Engineering 320 (2017) 237 – 260.

[14] M. E. Tipping, Sparse Bayesian learning and the relevance vector machine, Journal of Machine Learning Research 1 (2001) 211–244.

[15] M. E. Tipping, A. C. Faul, Fast marginal likelihood maximization for sparse Bayesian models, in: AISTATS, 2003.

[16] S. D. Babacan, R. Molina, A. K. Katsaggelos, Bayesian compressive sensing using laplace priors, IEEE Transactions on Image Processing 19 (2010) 53–63.

[17] M. Khalil, A. Sarkar, S. Adhikari, Tracking noisy limit cycle oscillation with nonlinear filters, Journal of Sound and Vibration 329 (2010) 150–170.

[18] Y. M. Marzouk, H. N. Najm, L. A. Rahn, Stochastic spectral methods for efficient bayesian solution of inverse problems, Journal of Computational Physics 224 (2007) 560 – 586.

[19] Y. M. Marzouk, H. N. Najm, Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems, Journal of Computational Physics 228 (2009) 1862 – 1902.

[20] T. Cui, Y. M. Marzouk, K. E. Willcox, Data-driven model reduction for the Bayesian solution of inverse problems, International Journal for Numerical Methods in Engineering 102 (2015) 966–990.

[21] D. Galbally, K. Fidkowski, K. Willcox, O. Ghattas, Non-linear model reduction for uncertainty quantification in large-scale inverse problems, International Journal for Numerical Methods in Engineering 81 (2010) 1581–1608.

[22] C. M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[23] A. C. Faul, M. E. Tipping, Analysis of sparse Bayesian learning, in: Advances in Neural Information Processing Systems 14, MIT Press, 2001, pp. 383–389.

[24] G. O. Roberts, A. Gelman, W. R. Gilks, Weak convergence and optimal scaling of random walk Metropolis algorithms, The Annals of Applied Probability 7 (1997) 110–120.

[25] W. Gilks (Ed.), S. Richardson (Ed.), D. Spiegelhalter (Ed.), P. Van der Heijden, B. Morgan, N. Keiding, Markov Chain Monte Carlo in Practice, Chapman and Hall/CRC, 1996.

[26] J. Ching, Y. Chen, Transitional Markov Chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging, Journal of Engineering Mechanics 133 (2007) 816–832.

[27] B. W. Silverman, Density Estimation for Statistics and Data Analysis, Chapman & Hall, London, 1986.

[28] C. L. Pettit, D. K. Wilson, Variational inference of cluster-weighted models for local and global sensitivity analysis, International Journal of Reliability and Safety 8 (2014) 196–227.

[29] J. Nocedal, S. J. Wright, Numerical Optimization, second ed., Springer, New York, NY, USA, 2006.

[30] A. R. Conn, N. I. M. Gould, Ph. L. Toint, Trust-Region Methods, SIAM, Philadelphia, PA, USA, 2000.

[31] E. Jones, T. Oliphant, P. Peterson, et al., SciPy: Open source scientific tools for Python, 2001–. URL: `http://www.scipy.org/`.

[32] T. W. Anderson, An introduction to multivariate statistical analysis, 3rd ed ed., Hoboken, NJ John Wiley, 2003. URL: `http://openlibrary.org/books/OL22543036M`.

[33] R. G. Ghanem, P. D. Spanos, Stochastic Finite Elements: A Spectral Approach, Springer-Verlag, Berlin, Heidelberg, 1991.

[34] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, S. Tarantola, Global Sensitivity Analysis: The Primer, Wiley, 2008. URL: `https://books.google.ca/books?id=wAssmt2vumgC`.

[35] D. W. Scott, Multivariate Density Estimation: Theory, Practice, and Visualization, John Wiley & Sons, New York, Chicester, 1992.

[36] K. B. Petersen, M. S. Pedersen, The matrix cookbook, 2012. URL: `http://www2.imm.dtu.dk/pubdb/p.php?3274`, version 20121115.

[37] D. Xiu, G. E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, SIAM J. Sci. Comput. 24 (2002) 619–644.

[38] N. Wiener, The Homogeneous Chaos, American Journal of Mathematics 60 (1938) 897–936.

[39] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, Rel. Eng. & Sys. Safety 93 (2008) 964–979.

Conflict of interest:

We have no conflict of interest to report.

Declaration of interest:
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Credit Author Statement

for the paper titled

Nonlinear sparse Bayesian learning for physics-based models

by R. Sandhu, C. Pettit, M. Khalil, D. Poirel and A. Sarkar.

All co-authors conceptualised, designed research and edited the paper, RS wrote software and the paper, and CP, MK, DP and AS mentored RS.