Proceedings of the ASME 2021
International Design Engineering Technical Conferences
and Computers and Information in Engineering Conference
IDETC/CIE2021
August 17 – 20, 2021, Online, Virtual

DETC2021-71570

DATA-DRIVEN DESIGN VIA SCALABLE GAUSSIAN PROCESSES FOR MULTI-RESPONSE BIG DATA WITH QUALITATIVE FACTORS

Liwei Wang [1,2], Suraj Yerramilli [3], Akshay Iyer [2], Daniel Apley [3], Ping Zhu [1], Wei Chen [2*]

[1] The State Key Laboratory of Mechanical System and Vibration, Shanghai Key Laboratory of Digital Manufacture for Thin-Walled Structures, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, P.R. China
[2] Dept. of Mechanical Engineering, Northwestern University, Evanston, IL, USA
[3] Dept. of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, USA

ABSTRACT

Scientific and engineering problems often require an inexpensive surrogate model to aid understanding and the search for promising designs. While Gaussian processes (GP) stand out as easy-to-use and interpretable learners in surrogate modeling, they have difficulties in accommodating big datasets, qualitative inputs, and multi-type responses obtained from different simulators, which has become a common challenge for a growing number of data-driven design applications. In this paper, we propose a GP model that utilizes latent variables and functions obtained through variational inference to address the aforementioned challenges simultaneously. The method is built upon the latent variable Gaussian process (LVGP) model where qualitative factors are mapped into a continuous latent space to enable GP modeling of mixed-variable datasets. By extending variational inference to LVGP models, the large training dataset is replaced by a small set of inducing points to address the scalability issue. Output response vectors are represented by a linear combination of independent latent functions, forming a flexible kernel structure to handle multi-type responses. Comparative studies demonstrate that the proposed method scales well for large datasets with over 10^4 *data points, while outperforming state-of-the-art machine learning methods without requiring much hyperparameter tuning. In addition, an interpretable latent space is obtained to draw insights into the effect of qualitative factors, such as those associated with "building blocks" of architectures and element choices in metamaterial and materials design. Our approach is demonstrated for machine learning of ternary oxide materials and topology optimization of a multiscale compliant mechanism with aperiodic microstructures and multiple materials.*

Keywords: Gaussian process, Surrogate modeling, Big data, Qualitative factor, Multi-type response, Latent variable, Topology optimization

1. INTRODUCTION

Spurred by the growth in computation capability and data resources, surrogate modeling is increasingly becoming an indispensable tool to expedite a design process and facilitate knowledge discovery in scientific and engineering problems [1]. Gaussian processes (GPs) have come to prevail in the arena of surrogate modeling with a wide range of applications in engineering designs, such as emulating responses of expensive simulations [2], model calibration [3], sensitivity analysis and uncertainty quantification [4]. However, Gaussian processes have limitations when applied to complex design problems with challenging characteristics, such as large datasets, qualitative design variables, and multi-type responses. Multiscale metamaterial systems design is such an example. It requires numerous on-the-fly homogenization calculations for each new metamaterial system design due to a large number of unit cells (microstructures) considered and the nested iterations in multiscale design. In this case, data-driven design methods can greatly accelerate the design process by using an inexpensive surrogate model to replace the costly on-the-fly homogenization [5]. However, the design of such systems often involves qualitative variables, such as the type of microstructure configurations and the choice of constituent materials [6, 7], that span an enormous combinatorial design space which can easily lead to exponential growth in the size of the database. Meanwhile, homogenized properties of interest for these metamaterials, e.g., stiffness tensors and thermal expansion

* Corresponding Author. Email: weichen@northwestern.edu

coefficients, are examples of multi-type responses with very different physical implications and units, lacking obvious distance metrics to describe their discrepancies. There is a need to extend GP modeling to address the obstacles caused by big data, qualitative inputs, and multi-type responses.

Progress has been made in the literature to address each of these three challenges separately. To accommodate big data, various scalable GPs have been proposed [8]. Depending on the nature of the approximations, they can be broadly classified into two types. Globally approximated models focus on constructing an approximated covariance matrix with lower complexity and storage requirement by selecting a subset of the training data [9, 10], discarding uncorrelated entries to form a sparse covariance matrix [11], or employing some reduced-rank structures for the covariance matrix [12]. In contrast, locally approximated models deploy the divide-and-conquer strategy by considering only a subset of training data in the neighborhood of the query point to compute the predictions [13]. Meanwhile, to handle qualitative factors and multi-type responses, various modified covariance structures have been proposed. For example, levels of qualitative variables are viewed as different responses with simplified covariance structures [14, 15] while non-separable covariance structures are devised to describe multi-type responses [16-19].

Recently, attempts have been made to simultaneously accommodate big data and multi-type outputs [20]. However, it is not straightforward to extend these methods to handle qualitative factors since the existing frameworks for qualitative inputs are usually incompatible with those for big data or multi-type outputs. For example, the locally approximated GP requires a distance metric defined in the input variable space to obtain a subset around the query point. However, defining appropriate distance metrics for qualitative input spaces is challenging. Therefore, to the best of the authors' knowledge, no existing method can simultaneously address all three challenges.

In this study, we propose a scalable latent variable GP (LVGP) modeling approach that can simultaneously accommodate large data sets, qualitative factors, and multi-type outputs. Specifically, as shown in Figure 1, the proposed model integrates three GP variants to handle each of the challenges, respectively, under one unified latent-variable framework [21]. First, we adopt our previously proposed LVGP model to handle qualitative variables [5, 22, 23] by mapping them into a continuous latent space to capture their joint effects on the responses. Second, to address the challenge of big data, a sparse variational (SV) approach is employed to replace the large dataset with sparse underlying inducing points to significantly reduce computation and storage complexity [24]. Finally, we model the multi-type outputs using a combination of independent latent functions, which is known as the linear model of coregionalization (LMC) [18].

The above three GP variants are combined into one unified GP modeling framework for large datasets with qualitative inputs and multi-type responses. While large data GP modeling for multi-response problems has been achieved using variational LMC [20], our contribution lies in extending the sparse variational concept to LVGP by defining inducing points in the latent space. Additionally, we propose two latent space structures for extending the LMC model to LVGP. The new synthesized GP model from this work has the following desirable features:

- **Generalizability**: Conventional correlation functions for GP modeling of continuous quantitative inputs can be readily applied to the dataset with qualitative factors by using the latent variable representation. The model is also flexible for accommodating multi-type responses.
- **Scalability**: The model can easily handle a large dataset with $n = 10^4 \sim 10^5$ data points in our case studies, reducing the complexity from $O(n^3)$ to $O(n_I^3)$ with the number of induing points $n_I \ll n$.
- **Accuracy**: We demonstrate in our study that the proposed model outperforms some of the state-of-the-art machine learning models, such as neural networks and boosted trees [25].
- **Interpretability**: A highly interpretable latent space of qualitative variables obtained from the proposed approach provides substantial insights into the black-box problem.

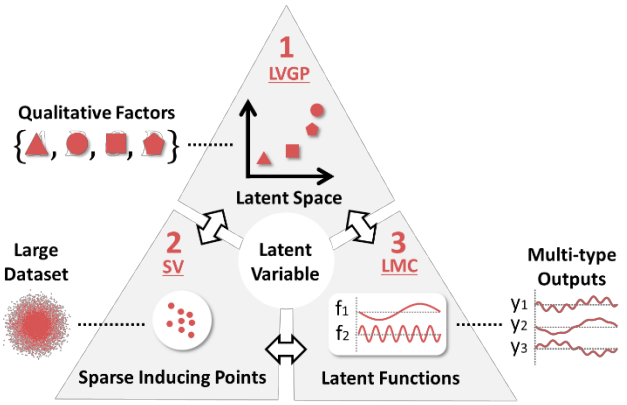

FIGURE 1: Three aspects integrated into the proposed Gaussian process model.

This synthesized GP model is useful for a wide range of data-driven engineering design applications that involve a combinatorial design space with mixed variables and multi-type responses. The aforementioned multiscale metamaterial system is such an example of complex engineering designs, which will be demonstrated in our case studies. Other possible applications include the discovery of new molecules with different combinations of atoms and the design of composite components with various choices of architectures and constituents that result in combinational search over mixed (qualitative and quantitative) variables.

The remaining paper is organized as follows. In Section 2, we provide a brief overview of the conventional Gaussian process modeling and explain its limitations with a large dataset with qualitative factors and multi-type outputs. Three aspects of the proposed approach are described in Section 3 by presenting three corresponding GP variants. Integration of these variants in developing a synthesized GP model is presented in Section 4. In Section 5, to validate the effectiveness, we compare the proposed method with some state-of-the-art machine learning models on

two numerical examples, and two engineering examples: one on multi-response machine learning for ternary oxide materials, and another on the data-driven design for aperiodic metamaterial systems. We conclude in Section 6 and discuss the scope for future applications.

2. REVIEW OF GAUSSIAN PROCESS MODELING

In this section, we provide an overview of GP modeling and explain the challenges posed by mixed variables, large datasets, and multi-type responses. For a single-output computer simulation model $y(\boldsymbol{x})$ with only quantitative inputs $\boldsymbol{x} = \{x_1, x_2, \dots, x_p\} \in R^p$, we assume $y(\boldsymbol{x})$ is a realization of a stochastic process:

$$Y(\boldsymbol{x}) = \boldsymbol{h}^T(\boldsymbol{x})\boldsymbol{\beta} + G(\boldsymbol{x}), \tag{1}$$

where $\boldsymbol{h}(\boldsymbol{x})$ is the prior mean function comprised of a vector of pre-defined basis functions $\boldsymbol{h}(\boldsymbol{x}) = [h_1(\boldsymbol{x}), \dots, h_m(\boldsymbol{x})]^T$, $\boldsymbol{\beta} = [\beta_1, \dots, \beta_m]^T$ is a vector of unknown weights for basis functions and $G(\boldsymbol{x})$ is a stationary multivariate Gaussian process with its covariance function defined as

$$cov\big(G(\boldsymbol{x}), G(\boldsymbol{x}')\big) = \sigma^2 r(\boldsymbol{x}, \boldsymbol{x}'), \tag{2}$$

where σ^2 is the prior variance and $r(\cdot,\cdot)$ is the correlation function. Among numerous existing correlation functions, the Gaussian correlation function is commonly used:

$$r(\boldsymbol{x}, \boldsymbol{x}') = \exp\{-(\boldsymbol{x} - \boldsymbol{x}')^T \boldsymbol{\Phi} (\boldsymbol{x} - \boldsymbol{x}')\}, \tag{3}$$

where $\boldsymbol{\Phi} = diag(\boldsymbol{\phi})$ and $\boldsymbol{\phi} = \big[\phi_1, \phi_2, \dots, \phi_p\big]^T$ are scaling parameters to characterize the variability of the sample functions. The construction of a GP model requires estimating the hyper-parameters $\boldsymbol{\beta}$, $\boldsymbol{\phi}$ and σ^2 based on the size-n training dataset with input $\mathbf{X} = \big\{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(n)}\big\}^T$ and output $\boldsymbol{y} = \big\{y^{(1)}, y^{(2)}, \dots, y^{(n)}\big\}^T$. A common way to determine the GP model parameters is to find a point estimate via maximum likelihood estimation (MLE). Herein, we assume a constant prior mean function with $\boldsymbol{h^T(x)\beta} = \beta$ for the GP model. The corresponding log-likelihood can be given after ignoring the constants:

$$L_{ln}(\boldsymbol{\phi}, \beta, \sigma^2) = -\frac{1}{2}\ln|\boldsymbol{K}(\boldsymbol{\phi})| - \frac{1}{2\sigma^2}(\boldsymbol{y} - \mathbf{1}\beta)^T \cdot \boldsymbol{K}(\boldsymbol{\phi})^{-1} \cdot (\boldsymbol{y} - \mathbf{1}\beta), \tag{4}$$

where $\ln(\cdot)$ is the natural logarithm, $\mathbf{1}$ is an $n \times 1$ vector of ones, and $\boldsymbol{K}$ is the $n \times n$ covariance matrix with $K_{ij} = \sigma^2 r\big(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\big)$ for $i, j = 1, \dots, n$. The hyperparameters are estimated by maximizing (4). With these estimated hyperparameters $\hat{\sigma}^2, \hat{\beta}$, and $\widehat{\boldsymbol{\phi}}$, the prediction $\hat{y}(\boldsymbol{x}^*)$ at any $\boldsymbol{x}^*$ can be obtained as:

$$\hat{y}(\boldsymbol{x}^*) = \hat{\beta} + \boldsymbol{r^T K^{-1}}\big(\boldsymbol{y} - \mathbf{1}\hat{\beta}\big), \tag{5}$$

where $\boldsymbol{r}(\boldsymbol{x}^*) = [r\big(\boldsymbol{x}^*, \boldsymbol{x}^{(1)}\big), r\big(\boldsymbol{x}^*, \boldsymbol{x}^{(2)}\big), \dots, r\big(\boldsymbol{x}^*, \boldsymbol{x}^{(n)}\big)]^T$. The posterior covariance between the responses at the two given data points $\boldsymbol{x}^*$ and $\boldsymbol{x}'$ is obtained as:

$$cov(y^*, y') = \hat{\sigma}^2 r(\boldsymbol{x}^*, \boldsymbol{x}') - \boldsymbol{r}(\boldsymbol{x}^*)^T \boldsymbol{K}^{-1} \boldsymbol{r}(\boldsymbol{x}'), \tag{6}$$

For more detailed illustrations and implementation of the GP modeling, readers are referred to [26].

As discussed in Section 1, this conventional Gaussian process will encounter various obstacles when applied to a large dataset with qualitative inputs and multiple-type outputs. Firstly, existing correlation functions are devised for quantitative variables and fail to accommodate qualitative variables. For example, the correlation function in Equation (3) relies on a distance metric defined for input variables to describe the correlation between responses at different data points. However, discrete levels of qualitative inputs only serve as a nomenclature without any well-defined distance metric. Secondly, GP models suffer from prohibitive computational costs and storage requirements on large datasets, due to computing $\boldsymbol{K}^{-1}$ and $|\boldsymbol{K}|$ in Equation (4). The subsequent computational and storage complexities are $O(n^3)$ and $O(n^2)$, respectively. Thirdly, it is not trivial to extend this GP model for multi-type outputs obtained from simulators that jointly simulate different types of quantities [17]. While training an independent single-response GP model for each output is straightforward, it entails a time-consuming training process, especially for large datasets. Also, if the correlation between outputs is poorly captured, the GP model will result in a poor prediction power and inappropriate joint uncertainty representation.

3. VARIANTS OF GAUSSIAN PROCESSES FOR ADDRESSING DATA CHALLENGES

In this section, we discuss three GP variants to address the data challenges associated with qualitative factors, big data, and multi-type output, respectively. These variants are all built upon the concept of latent representation, including the LVGP model for handling qualitative inputs, a sparse variational GP (SVGP) model with inducing points for managing big data challenges, and a GP model with the linear model of coregionalization (LMC) to predict multi-type outputs. These three variants will be integrated to form the proposed scalable LVGP approach in the next section.

3.1 LVGP Model for Qualitative Factors

Different levels of qualitative variables lack a well-defined distance metric, which precludes the use of conventional kernels devised for quantitative variables. However, as illustrated in mapping A of Figure 2, for a physical model, there are always some underlying quantitative physical variables that explain the effects of any qualitative factor on the response(s). The space spanned by these (perhaps extremely high-dimensional) underlying physical variables induces a natural distance metric

between levels of the qualitative variables. Therefore, according to sufficient dimension reduction arguments [27, 28], we could assume a low-dimensional latent space to capture the joint effects of these underlying variables, as shown in mapping B of Figure 2. Based on this insight, we recently proposed an LVGP model to enable GP modeling for a dataset with qualitative inputs [5, 23].

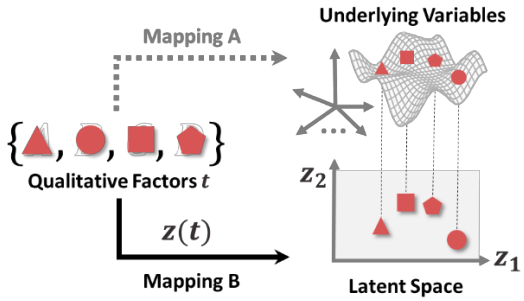

FIGURE 2: Illustration of the latent variable representation for qualitative factors. The shape is the qualitative factor here for geometry design.

Specifically, consider a single-response computer simulation model $y(\boldsymbol{u})$ with input $\boldsymbol{u} = [\boldsymbol{x}^T, \boldsymbol{t}^T]^T$ containing both quantitative variables $\boldsymbol{x} = [x_1, x_2, \dots, x_p]^T \in R^p$ and qualitative variables $\boldsymbol{t} = [t_1, t_2, \dots, t_q]^T$, with the j^{th} qualitative factor $t_j \in \{1,2 \dots, l_j\}$, where $l_j \in N^+$ is the total number of levels for t_j. By assuming a g-dimensional latent vector $\boldsymbol{z}_j(t_j) = [z_{j,1}(t_j), \dots, z_{j,g}(t_j)]^{\mathrm{T}} \in R^g$ for each t_j, the original mixed-variable input $\boldsymbol{x}$ can be transformed into quantitative input vector $\boldsymbol{s} = [\boldsymbol{x}^T, \boldsymbol{z}(\boldsymbol{t})^T]^T \in R^{p+q*g}$, where $\boldsymbol{z}(\boldsymbol{t}) = \left[\boldsymbol{z}_1(t_1)^T, \dots, \boldsymbol{z}_q(t_q)^T\right]^T$. The standard GP model can then be modified as (using a constant mean function):

$$Y(\boldsymbol{s}) = \beta + G(\boldsymbol{s}), \tag{7}$$

$$cov\big(G(\boldsymbol{s}), G(\boldsymbol{s}')\big) = \sigma^2 r(\boldsymbol{s}, \boldsymbol{s}'), \tag{8}$$

Since the transformed input vector $\boldsymbol{s}$ contains only quantitative variables, we can use any existing correlation function in equation (8). Herein, we still adopt the prevailing Gaussian correlation function:

$$r(\boldsymbol{s}, \boldsymbol{s}') = \exp\{-(\boldsymbol{x} - \boldsymbol{x}')^T \boldsymbol{\Phi} (\boldsymbol{x} - \boldsymbol{x}') - (\boldsymbol{z} - \boldsymbol{z}')^T \boldsymbol{\Phi}_{\boldsymbol{z}} (\boldsymbol{z} - \boldsymbol{z}')\}, \tag{9}$$

It should be noted that this correlation function contains two sets of parameters to be estimated: scaling parameters $\boldsymbol{\Phi}$ for quantitative variables and the set of latent vectors mapped from the qualitative variables $\boldsymbol{Z} = \cup_{i=1}^{q} \{\boldsymbol{z}_i(1), \dots, \boldsymbol{z}_i(l_i)\}$. The scaling parameter matrix $\boldsymbol{\Phi}_{\boldsymbol{z}}$ for latent variables is fixed to be an identity matrix in LVGP since these scaling factors are absorbed into the estimated latent variable values $\boldsymbol{Z}$. In our previous work [23], we follow the same procedure in Section 2 to estimate the values of β, σ^2, $\boldsymbol{\Phi}$, and $\boldsymbol{Z}$ via MLE.

LVGP enables easy integration with Bayesian optimization, which has been successfully applied in materials discovery and design [22, 29]. However, like conventional GPs, LVGPs also require enormous computation and storage resources when applied to big data. Moreover, the original LVGP could only accommodate a single response instead of multi-type responses. To address these, we need to integrate LVGP with the two GP variants introduced in the following subsections.

3.2 SVGP for Big Data

In this subsection, we introduce the concept of the sparse variational (SV) model where an *artificial* training dataset that is much smaller than the original training set is used to provide an approximately equivalent covariance information. These *artificial* training points, also called *inducing points*, might not be observed in the original training data and are not necessarily obtained from a real physical model. Instead, the locations and responses of these inducing points are estimated by stochastic variational inference [21] from the collected big data. This type of model is also called the sparse variational model [24].

Consider a large training dataset with quantitative input data $\mathbf{X} = [\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \dots, \boldsymbol{x}^{(n)}]^T$ and observed response data $\boldsymbol{y} = \left[y^{(1)}, y^{(2)}, \dots, y^{(n)}\right]^T$, where n is the size of the training data. In constructing the conventional GP model in Section 2, we assume a unified multivariate Gaussian distribution for the residual process $\boldsymbol{G}(\cdot)$ at n training input data points $\mathbf{X}$ and n_* query input data points $\mathbf{X}^*$:

$$\begin{bmatrix} \boldsymbol{G}(\mathbf{X}^*) \\ \boldsymbol{G}(\mathbf{X}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{G}_* \\ \boldsymbol{G} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \boldsymbol{K}_{**} & \boldsymbol{K}_{*\boldsymbol{X}} \\ \boldsymbol{K}_{*\boldsymbol{X}}^T & \boldsymbol{K}_{\boldsymbol{XX}} \end{bmatrix}\right), \tag{10}$$

where $\boldsymbol{K}_{*\boldsymbol{X}}$ is an $n_* \times n$ cross-covariance matrix between responses at $\mathbf{X}^*$ and $\mathbf{X}$, $\boldsymbol{K}_{**}$ is an $n_* \times n_*$ covariance matrix for $\mathbf{X}^*$, and $\boldsymbol{K}_{\boldsymbol{XX}}$ is an $n \times n$ covariance matrix for $\mathbf{X}$. $\boldsymbol{K}_{\boldsymbol{XX}}$ plays an essential role in both the training and prediction stages, as shown in Equations (4) through (6). We are using the covariance information of the training data stored in $\boldsymbol{K}_{\boldsymbol{XX}}$ to predict responses at $\mathbf{S}^*$. In other words, $\boldsymbol{G}_*$ at the query points $\mathbf{X}^*$ can be "*explained*" by $\boldsymbol{G}$ at the training points $\mathbf{X}$, as illustrated in the first row of Figure 3.

However, as discussed in Section 2, the use of this $n \times n$ covariance matrix $\boldsymbol{K}_{\boldsymbol{XX}}$ is the primary contributor to the curse of dimensionality in GP modeling. To address this issue, we assume that there is a small set of *inducing points* at the location $\mathbf{X}_{\boldsymbol{I}}$ ($n_I \ll n$), with the residual process $\boldsymbol{G}(\mathbf{X}_{\boldsymbol{I}})$ subjects to

$$\begin{bmatrix} \boldsymbol{G}(\mathbf{X}) \\ \boldsymbol{G}(\mathbf{X}_{\boldsymbol{I}}) \end{bmatrix} = \begin{bmatrix} \mathbf{G} \\ \mathbf{G}_{\boldsymbol{I}} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \boldsymbol{K}_{\boldsymbol{XX}} & \boldsymbol{K}_{\boldsymbol{XI}} \\ \boldsymbol{K}_{\boldsymbol{XI}}^T & \boldsymbol{K}_{\boldsymbol{II}} \end{bmatrix}\right), \tag{11}$$

as illustrated in the second row of Figure 3. Following the same logic in Equation (10), $\boldsymbol{G}$ at the size-n training dataset $\mathbf{X}$ can be "*explained*" by $\mathbf{G}_{\boldsymbol{I}}$ at the size-n_I inducing input data points $\mathbf{X}_{\boldsymbol{I}}$. The inducing points can now replace the original data to

improve efficiency. Under this setting, besides the original parameters in the LVGP model, we also need to estimate the locations and the corresponding $\mathbf{G_I}$ of these inducing points during the training process.

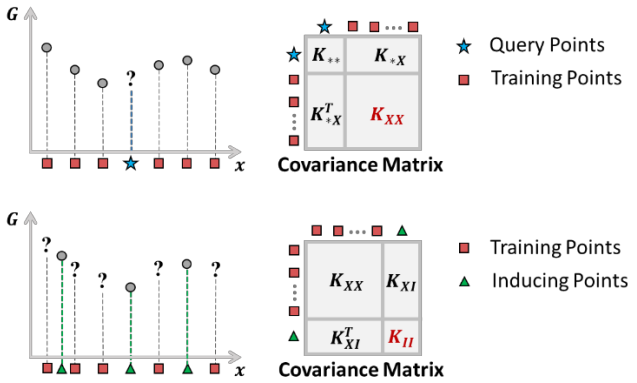

FIGURE 3: Covariance matrices used to describe residual process G at query points based on training points (first row) and describe G at the training points based on sparse inducing points (second row).

To achieve this, a variational distribution is defined to approximate the posterior:

$$q(\boldsymbol{G}, \boldsymbol{G_I}) = p(\boldsymbol{G}|\boldsymbol{G_I})q(\boldsymbol{G_I}), \tag{12}$$

where $q(\boldsymbol{G_I}) = \mathcal{N}(\boldsymbol{G_I}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the probability density function of the marginal variational distribution and $p(G|G_I)$ is the conditional distribution that is readily obtained from Equation (11). With these, parameters to be estimated include β, σ^2, $\boldsymbol{\Phi}$, $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, $\mathbf{X_I}$ and $\boldsymbol{G_I}$. Since maximizing the likelihood function will involve the costly calculation of $\boldsymbol{K_{XX}^{-1}}$ and $|\boldsymbol{K_{XX}}|$, we turn to estimate parameters by maximizing the evidence lower bound (ELBO):

$$ELBO = L_t - D_{KL}[q(\boldsymbol{G}, \boldsymbol{G_I})||p(\boldsymbol{G}, \boldsymbol{G_I})], \tag{13}$$

with the likelihood term L_t and the Kullback–Leibler (KL) divergence $D_{KL}[q(\boldsymbol{G}, \boldsymbol{G_I})||p(\boldsymbol{G}, \boldsymbol{G_I})]$ given as

$$L_t = \int log[p(\boldsymbol{y}|\boldsymbol{G})] \cdot \mathcal{N}(\boldsymbol{G}; \boldsymbol{A\mu}, \boldsymbol{A\Sigma A^T} + \boldsymbol{B}) d\boldsymbol{G}, \tag{14}$$

$$D_{KL}[q(\boldsymbol{G}, \boldsymbol{G_I})||p(\boldsymbol{G}, \boldsymbol{G_I})] = \frac{1}{2}\left\{log\left(\frac{|\boldsymbol{K_{II}}|}{|\boldsymbol{\Sigma}|}\right) - n_I\right\} + \frac{1}{2} tr\left(\boldsymbol{K_{II}^{-1}\Sigma}\right) + (\boldsymbol{0} - \boldsymbol{\mu})^T \boldsymbol{K_{II}^{-1}} (\boldsymbol{0} - \boldsymbol{\mu}), \tag{15}$$

where $\boldsymbol{A} = \boldsymbol{K_{IX}K_{II}^{-1}}$ and $\boldsymbol{B} = \boldsymbol{K_{XX}} - \boldsymbol{K_{XI}K_{II}^{-1}K_{XI}^T}$. From Equations (13) ~ (15), we note that the evaluation of ELBO does not involve the expensive calculation of $\boldsymbol{K_{XX}^{-1}}$ and $|\boldsymbol{K_{XX}}|$. Instead, it only requires $\boldsymbol{K_{II}^{-1}}$ and $|\boldsymbol{K_{II}}|$ with the calculation complexity reduced to $O(n_I^3)$. The storage requirement can be reduced to $O(n_b^2)$ by using mini-batch stochastic gradient descent algorithms, where $n_b \ll n$ is the size of mini-batch.

After the training, prediction at query points $\mathbf{X}^*$ can be readily obtained as

$$\hat{\boldsymbol{y}}(\mathbf{X}^*) = \hat{\beta} + \boldsymbol{K_{*I}K_{II}^{-1}}(\boldsymbol{\mu} - \boldsymbol{1}\hat{\beta}),$$
$$cov(\mathbf{X}^*, \mathbf{X}') = \left(\boldsymbol{K_{*I}K_{II}^{-1}}\right)\boldsymbol{\Sigma}\left(\boldsymbol{K_{*I}K_{II}^{-1}}\right)^T + \boldsymbol{K_{**}} - \boldsymbol{K_{*I}K_{II}^{-1}K_{*I}^T}, \tag{16}$$

The prediction in Equation (16) only depends on the sparse inducing points and thus remains efficient even with a large training data set. While this model only considers quantitative inputs, we extend it to accommodate datasets with mixed-variable inputs in Section 4.1.

3.3 LMC for Multi-type Responses

In this section, we introduce the linear model of coregionalization (LMC) approach to handle multi-type responses [18]. The key idea behind LMC is to represent a multivariate Gaussian process by a linear combination of independent univariate Gaussian processes. Consider a multi-response computer simulation model $\boldsymbol{y}(\boldsymbol{x})$ with output $\boldsymbol{y} = \left[y_1, y_2, \dots, y_{N_{op}}\right]^T \in R^{N_{op}}$. Assume the prior model for the outputs is constructed from a linear transformation $\boldsymbol{W} \in R^{N_{op} \times L}$ of L ($L \leq N_{op}$) independent latent functions $\boldsymbol{f}(\boldsymbol{x})$:

$$\boldsymbol{Y}(\boldsymbol{x}) = \boldsymbol{\beta} + \boldsymbol{G}(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{f}(\boldsymbol{x}), \tag{17}$$

where $\boldsymbol{\beta}$ is a vector of prior means, $\boldsymbol{G} = [G_1, G_2, \dots, G_{N_{op}}]^T$ is a multi-response stationary Gaussian process, $\boldsymbol{f}(\boldsymbol{x}) = \{f_l(\boldsymbol{x}_l)\}_{l=1}^L$ and $f_l(\boldsymbol{x}_l)$ is an independent Gaussian process with its covariance defined to be:

$$cov_l\left(f_l(\boldsymbol{x}_l), f_l(\boldsymbol{x}_l')\right) = \sigma^2 r_l(\boldsymbol{x}_l, \boldsymbol{x}_l'), \tag{18}$$

where $r_l(\cdot,\cdot)$ has the same definition as in Equation (3). By using this LMC structure, the covariance of multi-response stationary Gaussian process $\boldsymbol{G}$ is given by

$$cov\left(G_i(\boldsymbol{x}), G_j(\boldsymbol{x}')\right) = \sum_{l=1}^{L} W_{il} cov_l\left(f_l(\boldsymbol{x}_l), f_l(\boldsymbol{x}_l')\right) W_{jl}, \tag{19}$$

This can be written in matrix form as

$$\boldsymbol{K_{XX'}}\left(\mathbf{G}(\mathbf{X}), \mathbf{G}(\mathbf{X}')\right) = \sum_{l=1}^{L} \boldsymbol{K_{l,XX'}} \otimes \boldsymbol{T_l}, \tag{20}$$

where $\boldsymbol{T_l} = \boldsymbol{W_{:,l}W_{:,l}^T}$ with $\boldsymbol{W_{:,l}}$ being the l^{th} column of $\boldsymbol{W}$, $\otimes$ is the Kronecker product. To estimate parameters in the LMC model, we can follow a similar approach in Section 2 to obtain MLE (see [16] for the details).

4. SCALABLE MULTI-RESPONSE LATENT VARIABLE GAUSSIAN PROCESS

In this section, we illustrate how three GP variants are integrated for scalable multi-response latent variable Gaussian process modeling. We first extend the sparse variational inference to LVGP, enabling scalable modeling on a large dataset with qualitative factors. This sparse variational LVGP (SV-LVGP) is then generalized to multi-type responses by integrating LMC models with specially devised latent spaces of the qualitative variables.

4.1 Extension of Variational Inference to LVGP

As mentioned in Section 3.2, the essence of the SV model is to approximate the covariance information with a set of inducing points. In the LVGP model, the original inputs $\boldsymbol{u} = [\boldsymbol{x}^T, \boldsymbol{t}^T]^T$ with both quantitative $\boldsymbol{x}$ and qualitative factors $\boldsymbol{t}$ are transformed to quantitative inputs $\boldsymbol{s} = [\boldsymbol{x}^T, \boldsymbol{z}(\boldsymbol{t})^T]^T$ by mapping levels of qualitative factors $\boldsymbol{t}$ to the corresponding latent vectors $\boldsymbol{z}$. As a result, there are two different input spaces $\boldsymbol{u}$ and $\boldsymbol{s}$ that can be used to define the locations of inducing points, as illustrated in Figure 4.

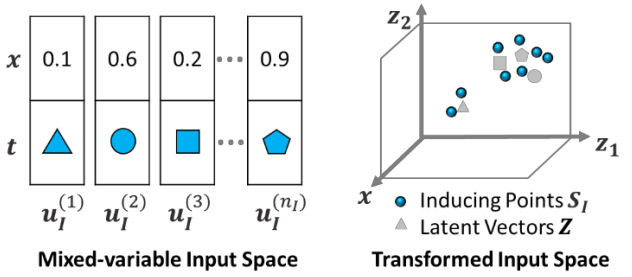

FIGURE 4: Defining the locations of inducing points in the mixed-variable input space (left) and the transformed quantitative input space (right).

For the former (shown in the left column of Figure 4), the variational inference process for the inducing points will become a mixed-variable optimization problem that is computationally expensive and sensitive to initialization. Therefore, we define the locations of inducing points in the transformed quantitative input space, as shown in the right column of Figure 4. We denote the locations of inducing points, transformed training points, and query input data points as $\mathbf{S_I}$, $\mathbf{S}$ and $\mathbf{S}^*$, respectively. The SVGP defined in Equations (10) ~ (16) can be introduced into LVGP by simply replacing $\mathbf{X_I}$, $\mathbf{X}$ and $\mathbf{X}^*$with $\mathbf{S_I}$, $\mathbf{S}$ and $\mathbf{S}^*$, respectively. The covariance matrices involved are calculated through Equations (8) and (9). We name this new integrated model as sparse variational latent variable GP (SV-LVGP). In SV-LVGP, parameters to be estimated include β, $\boldsymbol{\Phi}$, $\boldsymbol{Z}$, $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, $\mathbf{S_I}$ and $\boldsymbol{G_I}$, which can be obtained by maximizing the ELBO as discussed in Section 3.2. Note that $\boldsymbol{Z}$ and $\mathbf{S_I}$ are simultaneously optimized in the training process. The feasibility of this practice is grounded in the observation that these two parameters are coupled together in the covariance matrices involving inducing points in ELBO. For better estimation of the inducing points $\mathbf{S_I}$, we can fix the latent vectors $\boldsymbol{Z}$ in the later stages of optimization and optimize only $\mathbf{S_I}$. This SV-LVGP model can now accommodate a large data set with qualitative factors. It is highly scalable since the computational and storage complexity remain $O(n_I^3)$ and $O(n_b^2)$, respectively, with the number of induing points $n_I \ll n$.

4.2 Extension of LMC to SV-LVGP

In this subsection, we extend LMC to the proposed SV-LVGP model for multi-type responses and present two types of model structures (illustrated in Figure 5) - one with independent latent spaces and one with shared latent spaces. Specifically, to achieve this extension, the domain of latent functions in LMC is changed from the original mixed qualitative-quantitative input space to the transformed quantitative input space of SV-LVGP. In general cases, the qualitative variables might show different joint effects on different responses. Accordingly, we may construct an independent latent variable space for each latent function in LMC to capture different effects of qualitative variables, as shown in the first row of Figure 5.

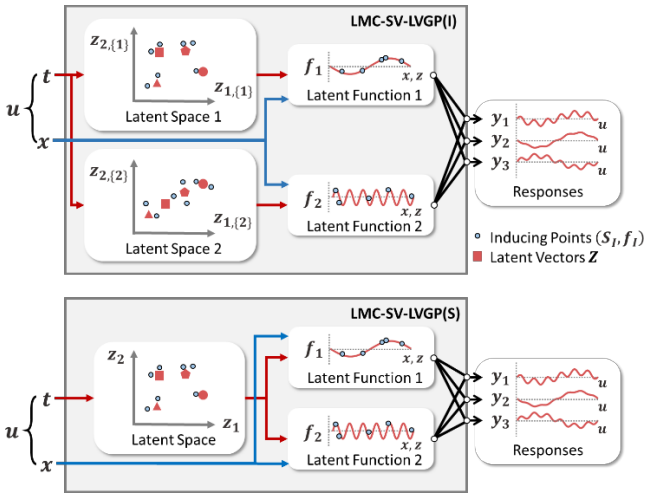

FIGURE 5: Illustration of the latent space structures for LMC-SV-LVGP(I) with independent latent spaces (first row) and LMC-SV-LVGP(S) with shared latent space (second row).

With this independent latent space structure, the original LMC model changed to

$$\boldsymbol{Y}(\boldsymbol{u}) = \boldsymbol{\beta} + \boldsymbol{G}(\boldsymbol{s}) = \boldsymbol{W}\boldsymbol{f}(\boldsymbol{s}), \tag{21}$$

where $\boldsymbol{s} = [\boldsymbol{x}^T, \boldsymbol{z}(\boldsymbol{t})^T]^T \in R^{p+L*q*g}$ is the mapped input corresponding to $\boldsymbol{u}$, $\boldsymbol{z}(\boldsymbol{t}) = \left[\boldsymbol{z}_1(t_1)^T, \dots, \boldsymbol{z}_q(t_q)^T\right]^T \in R^{L*q*g}$ is the assembled latent vector for all qualitative variables with $\boldsymbol{z}_i(t_i) = \left[\boldsymbol{z}_{i,\{1\}}(t_i)^T, \dots, \boldsymbol{z}_{i,\{L\}}(t_i)^T\right]^T \in R^{L*g}$, $\boldsymbol{z}_{i,\{l\}}(t_i) \in R^g$ is the latent vector of t_i for the l^{th} latent function, $\boldsymbol{f}(\boldsymbol{s}) = \{f_l(\boldsymbol{s}_l)\}_{l=1}^L$ with $\boldsymbol{s}_l = \left[\boldsymbol{x}^T, \boldsymbol{z}_{1,\{l\}}{}^T, \dots, \boldsymbol{z}_{q,\{l\}}{}^T\right]^T$. The definition of the correlation function $r_l(\cdot,\cdot)$ in (18) is changed to the one for LVGP as in Equation (9).

With these modifications, we could then follow the same procedure in Section 4.1 of introducing inducing points $(\boldsymbol{S_I}, \boldsymbol{G_I})$ for scalable GP modeling, but the computational complexity will surge to $O(N_{op}^3 n_I^3)$ due to the Kronecker product in Equation (20). To avoid this significant increase in the computational costs, we propose to use the values of the latent functions $\boldsymbol{f}(\boldsymbol{S_I})$ as the response data for the inducing points, instead of the final residual function $\boldsymbol{G}(\boldsymbol{S_I})$. $\boldsymbol{K_{II}}$ is now a block diagonal matrix, and the computational complexity is reduced to $O(Ln_I^3)$. Note that different latent functions in LMC will have different inducing points since they are defined on independent latent variable spaces. We will refer to this model as LMC-SV-LVGP(I).

In practice, one could impose constraints on the structure of the different latent variable spaces based on prior knowledge of the physical model to reduce the number of model parameters. For example, when the qualitative variables have similar joint effects on responses, higher efficiency and interpretability can be achieved by using a special structure shown in the second row of Figure 5, in which the latent functions share the same latent variable space for all the qualitative variables. Specifically, we modify the definition of the latent vector by setting $\boldsymbol{z}_{i,\{l\}}(t_i)^T \equiv \boldsymbol{z}_{i,\{1\}}(t_i)^T$ and $\boldsymbol{z}_i(t_i) = \left[\boldsymbol{z}_{i,\{1\}}(t_i)^T\right]^T \in R^g$. Moreover, we now estimate a different scaling parameter matrix $\boldsymbol{\Phi_z}$ for the latent variables (in Equation (9)) in different latent functions, instead of fixing them to be the identity matrix as was done earlier. These scaling parameters would account for small differences in the effects of the qualitative variables on the different responses. We refer to this variant with the shared latent variable space as LMC-SV-LVGP(S). Compared to the more general model with independent latent spaces, LMC-SV-LVGP(S) sacrifices some flexibility for improving optimization efficiency with fewer parameters and inducing points to be estimated. Moreover, in the case that the qualitative variables indeed have similar joint effects on different responses, the LMC-SV-LVGP(S) model will have comparable performance. We will highlight these trade-offs in the next section.

5. COMPARATIVE CASE STUDIES

To validate the effectiveness of our proposed methods, we compare them against two state-of-the-art machine learning models that are most commonly used for big data: neural networks (NN) [30] and the extreme gradient boosted decision trees (XGBoost) [25]. We include two numerical examples for numerical performance comparisons and two engineering problems to demonstrate the usefulness of the proposed methods in data-driven design, including machine learning of ternary oxide materials and topology optimization of a multiscale compliant mechanism. For all case studies, 10-fold cross-validation (CV) is performed for all the models to measure their predictive power. Note that the hyperparameters of the NN and XGBoost models are tuned in an additional CV process before the comparative validation to ensure the best performance. Specifically, a grid search and a random grid search are performed in the hyperparameter selection for NN and XGBoost, respectively, with the search space shown in Tables 1 and 2.

Table 1. The hyperparameter space of the grid search for NN

Number of hidden layers	Neurons per layer	Activation function	Learning rate
1, 2	4, 8, 16, 32, 64	'logistic', 'tanh', 'relu'	0.05, 0.01, 0.005, 0.001

Table 2. The range of the random grid search for XGBoost

Parameter	Range*	Parameter	Range
Colsample**	[0.3,0.7]	Learning rate	[0.03,0.3]
Gamma	(0.0,0.5]	Maximum depth	[2,6]
Number of estimators	[100,150]	Subsample***	[0.4,0.6]

* uniform distribution is assumed for each range.
** subsample ratio of columns when constructing each tree.
*** subsample ratio of the training instances.

In contrast, we intentionally avoid this exhaustive tuning process for all the proposed GP models to demonstrate their easy-of-use and generality. The proposed GP models are implemented using the GPflow package [31] in Python. The initial latent vectors for qualitative variables are randomly assigned while the locations of the initial inducing points are randomly selected from the training data. We use the natural gradient optimizer [32] to optimize the variational parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ while the Adam optimizer [33] is adopted for all other parameters for faster convergence and better parameter estimation [34, 35]. We train the GP models in batches of size 100 and set the maximum number of training iterations to 20,000.

5.1 Single-response Math Function

In this case study, we focus on a large single-response dataset with qualitative variables generated by a math function [36] given as

$$y = \begin{cases} 7sin(2\pi x_1 - \pi) + \quad sin(2\pi x_2 - \pi), \; if \; t = 1 \\ 7sin(2\pi x_1 - \pi) + \; 13sin(2\pi x_2 - \pi), \; if \; t = 2 \\ 7sin(2\pi x_1 - \pi) + 1.5sin(2\pi x_2 - \pi), \; if \; t = 3 \\ 7sin(2\pi x_1 - \pi) + 9.0sin(2\pi x_2 - \pi), \; if \; t = 4 \\ 7sin(2\pi x_1 - \pi) + 4.5sin(2\pi x_2 - \pi), \; if \; t = 5 \end{cases}, \quad (22)$$

where $x_1, x_1 \in [0,1]$ are continuous quantitative variables and $t \in \{1,2,3,4,5\}$ is a qualitative variable with five levels representing different coefficients for the second sine function. Therefore, the true ordering of different levels should be 1-3-5-4-2 based on the second coefficient. We generate a large dataset by sampling on a $100 \times 100 \times 5$ grid in the x_1- x_2-t space, rendering 50,000 data points. To test the sensitivity of the model, we consider Gaussian random noise with three different levels of standard derivation (SD), i.e., no noise (SD=0.0), low noise (SD=0.4), and high noise (SD=4.0). We adopt a 2D latent space

to represent the qualitative variable in SV-LVGP, which is reported in [23] to be sufficient for most physical problems. To study the influence of the number of inducing points, we trained a set of SV-LVGP models with 50, 100, and 500 inducing points, respectively. The performance is measured by root mean squared error (RMSE), as shown in Figure 6. In the ideal case, the RMSE value should be equal to the corresponding SD value of the noisy dataset.

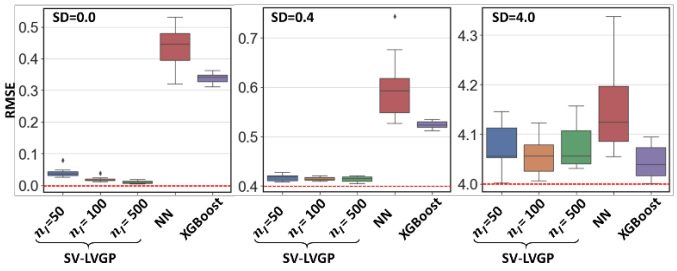

FIGURE 6: Boxplots of RMSE over 10-fold CV for all the models in the first case study under different noise levels. Red dashed lines represent the SD of the Gaussian noise.

In no noise (SD=0.0) and low noise (SD=0.4) situations, our SV-LVGP models outperform both NN and XGBoost, even with only 50 inducing points. The reason for the poor performance of NN and XGBoost may be due to the fact that the ordering of qualitative levels does not relate to their real underlying numerical values, eliminating a critical clue for the modeling. As the number of inducing points increases, so does the predictive power of SV-LVGP. However, when the level of the noise is high (SD=4.0), using a larger number of inducing points does not bring much benefit in prediction. Although XGBoost performs the best on the highly noisy dataset, the SV-LVGP model with 100 inducing points has a similar performance. Note that SV-LVGP models achieve this high accuracy without much tuning of the hyperparameters (which was done for NN and XGBoost). This demonstrates the robustness and ease-of-use of the SV-LVGP model.

Moreover, the proposed model provides interpretation for the levels of the qualitative variable through the latent space shown in Figure 7. It can be seen that the latent vectors mapped from different levels of the qualitative variable reside on a straight line with a correct ordering as 1-3-5-4-2. Thus, the correlation structure captured by this mapping agrees closely with the real underlying numerical values (the coefficient of the second sine function). Therefore, even though the correlation information is lost in the qualitative representation, it can be rediscovered from the data by using the proposed model. This could provide extra knowledge when applied to an unknown physical model. In contrast, NN does not have this interpretability while XGBoost fails to provide a quantitative measure for the correlation between classes. Moreover, it should be noted that the inducing points surround the latent vectors in the latent space. This is because all qualitative inputs in the training data are mapped to those latent vectors. As a result, regions around the latent vectors are the most critical to describe the statistical characteristics of training data. This provides another validation of the proposed method.

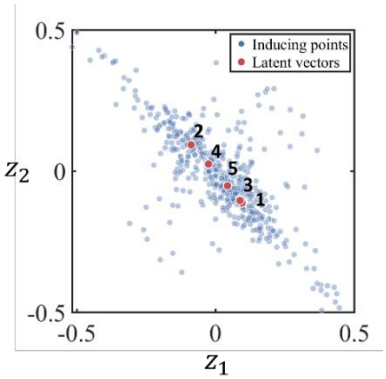

FIGURE 7: Latent vectors and inducing points in the latent space of SV-LVGP model with $n_I = 500$. The level of t corresponding to each latent vector is marked in the figure.

5.2 Multi-response Math Function

In this example, we use a mathematical multi-response dataset to validate the effectiveness of the LMC-SV-LVGP model. The corresponding multi-response math function is

$$y_1 = \sum_{i=1}^{2} \frac{x_i(t_{2-i}-3)}{80} + \prod_{j=1}^{2} cos\left(\frac{x_j}{\sqrt{j}}\right) cos\left(\frac{50(t_j-3)}{\sqrt{2}}\right)$$
$$y_2 = \sum_{i=1}^{2} \frac{x_i(t_{2-i}-3)}{80} \quad (23)$$
$$+ \prod_{j=1}^{2} cos\left(\frac{x_j}{\sqrt{j}} - \frac{(j-1)\pi}{2}\right) cos\left(\frac{50(t_j-3)}{\sqrt{2}}\right),$$

where y_1, y_2 are two responses, x_1, $x_1 \in [-100,100]$ are continuous quantitative variables and t_1, $t_2 \in \{1,2,3,4,5\}$ are qualitative variables with five levels. We generate a large dataset with 22,500 data points from a $30 \times 30 \times 5 \times 5$ uniform grid in the x_1-x_2-t_1-t_2 space. Similarly, we consider three levels of Gaussian noise for the dataset, i.e., SD=0.0, 0.1, 1.0. Both single-response SV-LVGP and multi-response LMC-SV-LVGP are considered in this case study. Specifically, we fit an independent SV-LVGP model for each output, which will be used as a reference for other multi-response models. For multi-response LMC-SV-LVGP, we consider three different structures: a. LMC-SV-LVGP(S) model with just a single latent function for the LMC kernel, which degenerates to the separable kernel [37], b. LMC-SV-LVGP(S) model with $L = 2$ latent functions for the LMC kernel, c. LMC-SV-LVGP(I) model with $L = 2$ latent functions for the LMC kernel. For all these models, 100 inducing points are used for the sparse variational inference. The performance of all the models over the 10-fold CV is given in Figure 8.

It can be noted that all three LMC-SV-LVGP models have lower average RMSE values than both NN and XGBoost. The more latent functions considered in the model, the better the performance of LMC-SV-LVGP. For this example, there is no significant difference between LMC-SV-LVGP models with shared or independent latent space, indicating the similar joint

effects of qualitative variables on the two responses. It is interesting to note that LMC-SV-LVGP models generally outperform the SV-LVGP model. In fact, SV-LVGP can be viewed as a special case of the LMC-SV-LVGP(I) model with the $\boldsymbol{W}$ matrix restricted to be a diagonal matrix. Therefore, a more flexible structure to exploit commonalities across responses should be the reason for the better performance of LMC-SV-LVGP model over its single-response counterpart. We show the latent spaces of LMC-SV-LVGP(S) and LMC-SV-LVGP(I) with $L = 2$ latent functions in Figure 9.

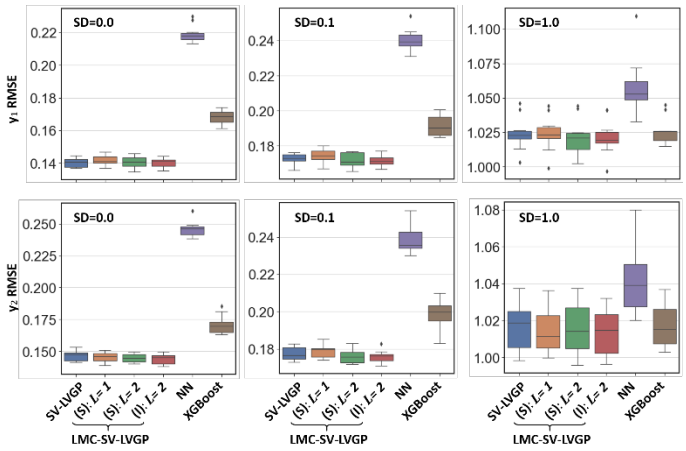

FIGURE 8: Boxplots of RMSE over 10-fold CV for all the models in the second case study under different noise levels. The first (second) row presents the result of the first (second) response.

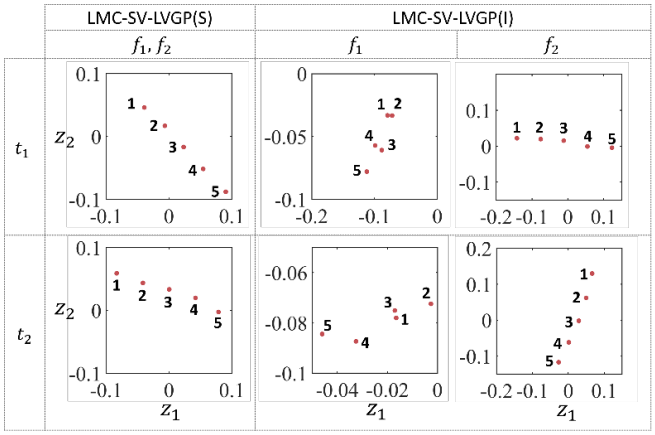

FIGURE 9: Latent space of LMCSV-LVGP(S) and LMC-SV-LVGP(I). The first (second) row presents the latent space for the first (second) qualitative variable.

For the LMC-SV-LVGP(S) model with a shared latent space for the two latent functions, different levels of the two qualitative variables are both equally distributed on a straight line and correctly ordered as 1-2-3-4-5, which again agrees with the underlying numerical t_1, t_2 in Equation (23). For the LMC-SV-LVGP(I) model, the two latent functions have independent latent space for the qualitative variables. In this case, while similar equally spacing latent points are observed for the second latent function, the latent embedding of qualitative variables for the first latent function has a very different pattern. The reason can be explained from the linear transformation for the latent functions with $\boldsymbol{W}$ learned from the training process:

$$\boldsymbol{Y}(\boldsymbol{u}) = \boldsymbol{W}\boldsymbol{f}(\boldsymbol{s}) = \begin{bmatrix} -0.02 & 1.14 \\ -0.03 & 1.12 \end{bmatrix} \cdot \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \qquad (24)$$

Note that the weights assigned to the second latent functions are much larger than that of the first latent function, which indicates that the second latent function dominates the prediction result. As a result, the latent space of the second latent function captured most of the correlation information between different levels of qualitative variables, implying a similar joint effect for the qualitative variables on the two responses. This shows how the latent space can help to extract knowledge on the input-output relations.

5.3 Machine Learning for Ternary Oxide Materials

Materials informatics require a surrogate model to replace the expensive simulation or experiments in accelerating high-throughput materials discovery and iterative design process [38]. In this case study, we demonstrate that the proposed method lends itself well for use in machine learning for the combinatorial design of materials composition, by applying it to predict both formation energy and stability of ternary oxide materials. Specifically, multi-response property data for 2030 ternary oxide materials have been extracted from the Open Quantum Material Database (OQMD) [39]. These ternary oxide materials have the molecular formula as $A_{x_1}B_{x_2}O_{x_3}$, where A and B can be selected from a set of 25 and 22 elements, respectively, and O is the oxygen atom. A and B are qualitative inputs, and x_1~x_3 are quantitative inputs, forming a mixed-variable input space for the model with the formation energy and the stability as outputs. Five models are trained on the dataset: a. SV-LVGP with 100 inducing points, b. LMC-SV-LVGP(S) model with $L = 2$ latent functions and 100 inducing points, c. LMC-SV-LVGP(I) model with $L = 2$ latent functions and 100 inducing points, d. NN and e. XGBoost. From their RRMSE values over 10-fold CV shown in Figure 10, it can be concluded that all three proposed models outperform both NN and XGBoost in predicting the formation energy and stability. Multi-response LMC-SV-LVGP models perform better than single-response SV-LVGP as before. There is a significant increase in performance when the shared space is replaced by independent latent spaces. This indicates that the type of elements included in A and B has different joint effects on the formation of energy and stability. The linear transformation for the latent functions in the LMC-SV-LVGP(I) model is

$$\begin{bmatrix} \text{formation energy} \\ \text{stability} \end{bmatrix} = \boldsymbol{W}\boldsymbol{f}(\boldsymbol{s}) = \begin{bmatrix} 1.41 & 0.08 \\ 0.80 & 1.12 \end{bmatrix} \cdot \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}. \qquad (25)$$

The first latent function f_1 dominates the prediction of formation energy while the second latent function f_2 contributes the most for the stability prediction, indicating a large

discrepancy between the two responses. We show the latent space of two qualitative variables in Figure 11, which contains rich information on the effects of element types.

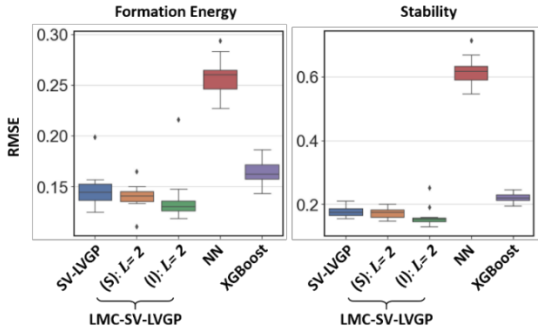

FIGURE 10: Boxplots of RMSE over 10-fold CV for all the models in the third case study. The left and right figures correspond to formation energy and stability, respectively.

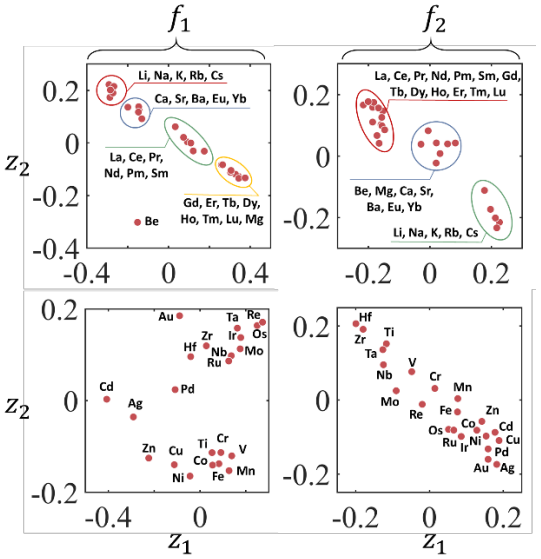

FIGURE 11: Latent space of LMC-SV-LVGP(I) trained on the ternary oxide materials dataset. The first (second) row shows the latent space for element A (B) in the molecular formula. The first (second) column shows the latent space used in the first (second) latent function.

For example, elements in position *A* form four clusters in the latent space for the first latent function. The majority of elements in each cluster belong to a specific element group in the periodic table, i.e., the alkali element group (marked by a red ellipse), the alkaline-earth element group (marked by a blue ellipse), the first and second half of the lanthanides element group (marked by a green and a yellow ellipse, respectively). Since f_1 dominates the prediction of formation energy, this clustering indicates that these groups have different effects on the formation energy. In contrast, there are only three clusters in the latent space for f_2, with the elements from the lanthanides element group being merged into the same cluster (marked by a red ellipse), indicating that all lanthanides elements have a similar influence on stability. Moreover, the proposed models require less time for the training and prediction after replacing the large data with 100 inducing points, thereby greatly reducing the time for high-throughput materials filtering or iterative design.

5.4 Data-driven Aperiodic Metamaterials System Design

In this case study, we demonstrate the usefulness of the proposed method in data-driven multiscale designs by applying it to a large database of unit-cell metamaterials for the design of aperiodic complex metamaterial systems [5, 40]. The microstructures are composed of two different base materials with one stiffer than the other. There are four variables to describe the microstructure of metamaterials, the volume fraction x of the stiff material, the class of microstructure t_1, the type of stiff material t_2 and the type of soft material t_3. $x \in [0,1]$ is a quantitative input for the surrogate model while t_1 through t_3 are qualitative inputs with the definition of their discrete levels shown in Figure 12. Large data is expected for such problems due to the high number of possible combinations.

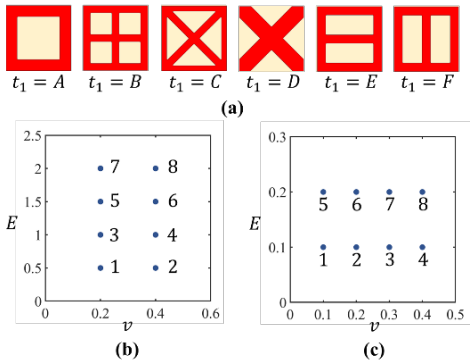

FIGURE 12: Qualitative variables of metamaterials. (a) Microstructure classes with red and yellow regions represent the stiff and soft base materials, respectively. (b) and (c) show Young's moduli and Poisson's ratios for different choices of the stiff material and the soft material, respectively.

We generated 19,200 microstructures with precomputed stiffness tensor by uniformly sampling 100 volume fraction values x for each possible combination of qualitative variables. The stiffness tensor is calculated through energy-based homogenization which takes 3 hours to compute for the whole database on a single CPU (Intel i7-9750H 2.6GHz). Note that this evaluation process is only performed once for the database construction but can be applied to numerous data-driven design cases. Independent entries of the stiffness tensor, i.e., C_{11}, C_{12}, C_{22} and C_{33}, are viewed as outputs for the surrogate model. SV-LVGP, LMC-SV-LVGP(S), LMC-SV-LVGP(I) with four latent functions, NN and XGBoost are trained on this metamaterial dataset to compare the predictive precision, as shown in Figure 13.

The three proposed models have much higher predictive power than both NN and XGBoost. While the single-response

SV-LVGP model has the best performance, the difference among the three proposed models is not so obvious. However, as demonstrated in [5], LMC-SV-LVGP(S) is more desirable in metamaterial system design due to a much lower dimensionality of the transformed design variables (a 7D vector). Moreover, the latent space of LMC-SV-LVGP(S) provides a highly interpretable distance metric for different qualitative variables, as shown in Figure 14, which will be very beneficial for the optimization process.

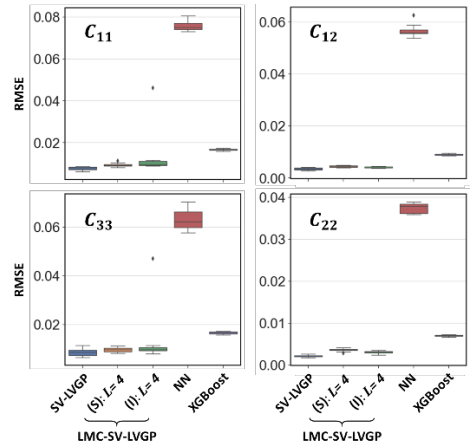

FIGURE 13: Boxplots of RMSE over 10-fold CV for all the models in the third case study. Each subfigure represents the result for an entry in the stiffness tensor.

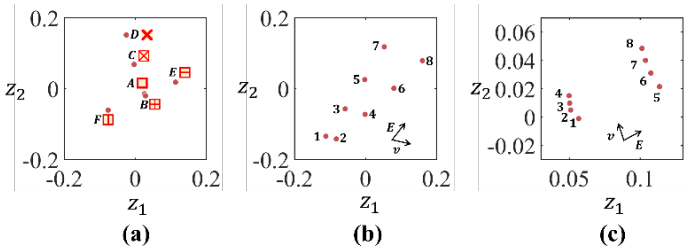

FIGURE 14: Latent space of LMCSV-LVGP(S) trained on the metamaterial database. (a) latent space of microstructure classes. (b) latent space of the stiff material. (c) latent space of the soft material.

Specifically, different classes of microstructures are distributed in a way that could reflect their similarity in the directional characteristics of the stiffness tensor considering multi-type responses. For example, classes A and B nearly overlap in the latent space shown in Figure 14 (a), which agrees with the fact that they have almost equivalent stiffness tensor under the homogenization assumption. Classes C and D are the closest neighbors for each other since they are the only pair with diagonal rods to resist shear strain. By comparing Figures 14 (b)~(c) with Figures 14 (b)~(c), it can be noted that the latent embeddings for the stiff and soft materials match well with the underlying values of Young's moduli and Poisson's ratios. We mark the two ascending directions for Young's modulus and Poisson's ratio in the latent space, respectively. Materials with similar Young's modulus are close to each other in the latent space. This indicates that Young's modulus has a larger impact on the stiffness tensor than Poisson's ratio. To demonstrate the usefulness of the proposed method in the multiscale metamaterial systems design, we apply it in designing a multiscale compliant mechanism [41], as shown in Figure 15 (a).

Consider a linear strained based actuator acting on the component, which can be modeled as a spring with stiffness $k = 0.1$ and a force $F_{in} = 1$. We aim to maximize the displacement u_{out} performed on a workpiece modeled by a spring with stiffness k through designing both macro- and microscale configurations. The design region is discretized into a 60×40 coarse mesh with each element filled by a microstructure discretized into a 200×200 finer mesh. The constraints imposed on the volume fraction of the stiff and soft materials are 0.3 and 0.1, respectively. Each coarse element is associated with the 7D transformed input vector as microscale design variables, i.e., the volume fraction x of the stiff material and three sets of 2D latent vectors for the class of microstructure t_1, the type of stiff material t_2 and the type of soft material t_3, respectively. Each coarse element also has a macroscale topological design variable $\rho \in [0,1]$ with zero and one representing void and solid, respectively. Therefore, we only need an 8D design vector to represent the complex macro- and microscale configurations for each coarse element. In contrast, the conventional TO framework uses one-hot encoding to represent the three qualitative variables, resulting in a 23D design vector for each element, i.e., one macroscale topological design variable ρ, 6D one-hot encoding for the class of microstructure t_1, and two sets of 8D one-hot encoding for the type of stiff material t_2 and the type of soft material t_3, respectively. Moreover, the dimension of the design variables will increase when more microstructure classes and materials are considered, while the design variables in our framework remain the same. This demonstrates the usefulness of the latent representation for the qualitative variables in reducing the dimension of design variables.

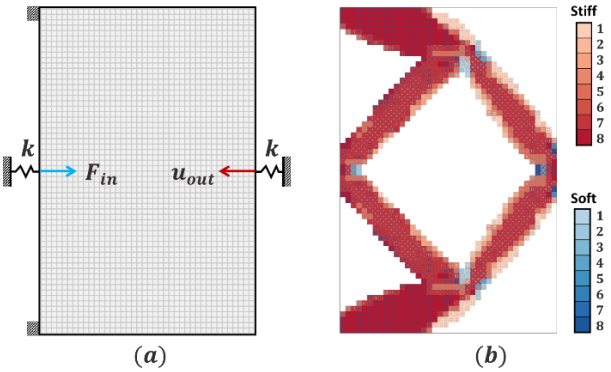

FIGURE 15: (a) Problem setting and (b) optimized mechanism, different types of stiff and soft materials are marked by red and blue gradient colormaps, respectively.

With the above definition, we follow the multi-scale TO framework proposed in [5] to optimize the macro-structure, the microscale configurations, and constituent materials

simultaneously. Specifically, in each iteration, the proposed LMC-SV-LVGP(S) model provides the homogenized stiffness tensor and its gradient with respect to microscale design variables for each coarse element. The method of moving asymptotes [42] is then adopted to iteratively optimize the design variables based on the sensitivity value. After the optimization, the optimized multiscale design is obtained, which increases u_{out} from 0.0558 to 1.3639, as shown in Figure 15 (b). Note that all eight classes of microstructures are used in the optimized structure, aligning in a way that matches with the main load-bearing directions of the macrostructure. The joint regions of different macroscale rods are composed of very soft materials, serving as hinges for the mechanism. This demonstrates the effectiveness of the simultaneous exploration of microscale configurations as well as constituent materials. In contrast, the periodic design obtained by using the same microscale design variables for all coarse elements generates a much smaller output displacement $u_{out} = 0.8147$, highlighting the advantages of aperiodic design. Moreover, due to the low-dimensional latent variables and inexpensive LVGP model, the overall design process only takes 253 iterations and less than two minutes to converge even with 96 million fine elements in the FEA model. In contrast, the conventional aperiodic multi-scale TO needs more iterations to converge and requires around 22 minutes for the on-the-fly homogenization process alone in each optimization iteration. This demonstrates that the use of the proposed surrogate model greatly accelerates the multiscale design process featuring a large combinatorial design space.

6. CONCLUSIONS

In this work, we have proposed a novel GP modeling approach that can accommodate big data with qualitative factors and multi-type responses. The proposed model integrates three modules based on the concept of latent variables, which has been highlighted in this work as a powerful tool to reduce computation complexity while increasing generality and interpretability. To address the big data challenge for problems with qualitative factors, we have first proposed the SV-LVGP model, which extends sparse variational inference to the LVGP for scalable modeling by using inducing points. The SV-LVGP model is further generalized to cases with multi-type responses by integrating the linear model of coregionalization with special latent space structures. Comparative studies demonstrate that the proposed model can easily handle $10^4 \sim 10^5$ training data points and achieve a high prediction performance that can compete with, and in most of the cases exceed, that of the state-of-the-art machine learning methods such as NN and XGBoost. The proposed model is also much easier to fit compared with these latter counterparts because it does not require a significant tuning effort. Moreover, we can gain considerable insights into the joint effects of qualitative variables on the responses based on the highly interpretable latent variable space. The most remarkable demonstration of this interpretability comes from the case study for ternary oxide materials, where clusters in the latent space relate to different element groups. This differentiates our method from other conventional black-box machine learning models. Through designing a compliant mechanism, we demonstrate that the design of multiscale metamaterial systems is greatly accelerated by using the data-driven approach and the proposed LVGP model that surrogates the material law of unit-cell structures. These promising results indicate that our method can be a useful tool to expedite designs where a large number of levels are associated with each qualitative variable or the design solutions are combinatorial in nature, such as the automated design and discovery in emerging material systems.

ACKNOWLEDGEMENTS

Support from the National Science Foundation (NSF) (Grant No. OAC 1835782) is greatly appreciated. Mr. Liwei Wang would like to acknowledge the support from the Zhiyuan Honors Program in Shanghai Jiao Tong University for his predoctoral study at Northwestern University.

REFERENCES

[1] Forrester, A., Sobester, A., and Keane, A., 2008, Engineering design via surrogate modelling: a practical guide, John Wiley & Sons.

[2] Tao, S., Shintani, K., Bostanabad, R., Chan, Y.-C., Yang, G., Meingast, H., and Chen, W., "Enhanced Gaussian process metamodeling and collaborative optimization for vehicle suspension design optimization," Proc. ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers Digital Collection.

[3] Gardner, P., Rogers, T., Lord, C., and Barthorpe, R., 2021, "Learning model discrepancy: A Gaussian process and sampling-based approach," Mechanical Systems and Signal Processing, 152, p. 107381.

[4] Bostanabad, R., Liang, B., Gao, J., Liu, W. K., Cao, J., Zeng, D., Su, X., Xu, H., Li, Y., and Chen, W., 2018, "Uncertainty quantification in multiscale simulation of woven fiber composites," Computer Methods in Applied Mechanics and Engineering, 338, pp. 506-532.

[5] Wang, L., Tao, S., Zhu, P., and Chen, W., 2021, "Data-Driven Topology Optimization With Multiclass Microstructures Using Latent Variable Gaussian Process," Journal of Mechanical Design, 143(3).

[6] Bauer, J., Meza, L. R., Schaedler, T. A., Schwaiger, R., Zheng, X., and Valdevit, L., 2017, "Nanolattices: an emerging class of mechanical metamaterials," Advanced Materials, 29(40), p. 1701850.

[7] Momeni, K., Mofidian, S. M., and Bardaweel, H., 2019, "Systematic design of high-strength multicomponent metamaterials," Materials & Design, 183, p. 108124.

[8] Liu, H., Ong, Y.-S., Shen, X., and Cai, J., 2020, "When Gaussian process meets big data: A review of scalable GPs," IEEE transactions on neural networks and learning systems, 31(11), pp. 4405-4423.

[9] Bostanabad, R., Chan, Y.-C., Wang, L., Zhu, P., and Chen, W., 2019, "Globally approximate gaussian processes for big data with application to data-driven metamaterials design," Journal of Mechanical Design, 141(11).

[10] Chalupka, K., Williams, C. K., and Murray, I., 2013, "A framework for evaluating approximation methods for Gaussian process regression," Journal of Machine Learning Research, 14, pp. 333-350.
[11] Gneiting, T., 2002, "Compactly supported correlation functions," Journal of Multivariate Analysis, 83(2), pp. 493-508.
[12] Wilson, A., and Nickisch, H., "Kernel interpolation for scalable structured Gaussian processes (KISS-GP)," Proc. International Conference on Machine Learning, PMLR, pp. 1775-1784.
[13] Gramacy, R. B., and Apley, D. W., 2015, "Local Gaussian process approximation for large computer experiments," Journal of Computational and Graphical Statistics, 24(2), pp. 561-578.
[14] Deng, X., Lin, C. D., Liu, K.-W., and Rowe, R., 2017, "Additive Gaussian process for computer models with qualitative and quantitative factors," Technometrics, 59(3), pp. 283-292.
[15] Qian, P. Z. G., Wu, H., and Wu, C. J., 2008, "Gaussian process models for computer experiments with qualitative and quantitative factors," Technometrics, 50(3), pp. 383-396.
[16] Alvarez, M. A., Rosasco, L., and Lawrence, N. D., 2011, "Kernels for vector-valued functions: A review," arXiv preprint arXiv:1106.6251.
[17] Fricker, T. E., Oakley, J. E., and Urban, N. M., 2013, "Multivariate Gaussian process emulators with nonseparable covariance structures," Technometrics, 55(1), pp. 47-56.
[18] Gelfand, A. E., Schmidt, A. M., Banerjee, S., and Sirmans, C., 2004, "Nonstationary multivariate process modeling through spatially varying coregionalization," Test, 13(2), pp. 263-312.
[19] Higdon, D., 2002, "Space and space-time modeling using process convolutions," Quantitative methods for current environmental issues, Springer, pp. 37-56.
[20] van der Wilk, M., Dutordoir, V., John, S., Artemev, A., Adam, V., and Hensman, J., 2020, "A framework for interdomain and multioutput Gaussian processes," arXiv preprint arXiv:2003.01115.
[21] Barber, D., 2012, Bayesian reasoning and machine learning, Cambridge University Press.
[22] Zhang, Y., Apley, D. W., and Chen, W., 2020, "Bayesian optimization for materials design with mixed quantitative and qualitative variables," Scientific reports, 10(1), pp. 1-13.
[23] Zhang, Y., Tao, S., Chen, W., and Apley, D. W., 2020, "A latent variable approach to Gaussian process modeling with qualitative and quantitative factors," Technometrics, 62(3), pp. 291-302.
[24] Hensman, J., Fusi, N., and Lawrence, N. D., 2013, "Gaussian processes for big data," arXiv preprint arXiv:1309.6835.
[25] Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., and Cho, H., 2015, "Xgboost: extreme gradient boosting," R package version 0.4-2, 1(4).
[26] Rasmussen, C. E., Williams, C. K. I., Press, M. I. T., Bach, F., and ProQuest, 2006, Gaussian Processes for Machine Learning, MIT Press.
[27] Cook, R. D., and Ni, L., 2005, "Sufficient dimension reduction via inverse regression: A minimum discrepancy approach," Journal of the American Statistical Association, 100(470), pp. 410-428.
[28] Li, K.-C., 1991, "Sliced inverse regression for dimension reduction," Journal of the American Statistical Association, 86(414), pp. 316-327.
[29] Wang, Y., Iyer, A., Chen, W., and Rondinelli, J. M., 2020, "Featureless adaptive optimization accelerates functional electronic materials design," Applied Physics Reviews, 7(4), p. 041403.
[30] LeCun, Y., Bengio, Y., and Hinton, G., 2015, "Deep learning," nature, 521(7553), pp. 436-444.
[31] Matthews, A. G. d. G., Van Der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J., 2017, "GPflow: A Gaussian Process Library using TensorFlow," J. Mach. Learn. Res., 18(40), pp. 1-6.
[32] Honkela, A., Raiko, T., Kuusela, M., Tornio, M., and Karhunen, J., 2010, "Approximate Riemannian conjugate gradient learning for fixed-form variational Bayes," The Journal of Machine Learning Research, 11, pp. 3235-3268.
[33] Kingma, D. P., and Ba, J., 2014, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980.
[34] Hensman, J., Rattray, M., and Lawrence, N. D., 2012, "Fast variational inference in the conjugate exponential family," arXiv preprint arXiv:1206.5162.
[35] Salimbeni, H., Eleftheriadis, S., and Hensman, J., "Natural gradients in practice: Non-conjugate variational inference in gaussian process models," Proc. International Conference on Artificial Intelligence and Statistics, PMLR, pp. 689-697.
[36] Swiler, L. P., Hough, P. D., Qian, P., Xu, X., Storlie, C., and Lee, H., 2014, "Surrogate models for mixed discrete-continuous variables," Constraint Programming and Decision Making, Springer, pp. 181-202.
[37] Conti, S., Gosling, J. P., Oakley, J. E., and O'Hagan, A., 2009, "Gaussian process emulation of dynamic computer codes," Biometrika, 96(3), pp. 663-676.
[38] Kailkhura, B., Gallagher, B., Kim, S., Hiszpanski, A., and Han, T. Y.-J., 2019, "Reliable and explainable machine-learning methods for accelerated material discovery," npj Computational Materials, 5(1), pp. 1-9.
[39] Kirklin, S., Saal, J. E., Meredig, B., Thompson, A., Doak, J. W., Aykol, M., Rühl, S., and Wolverton, C., 2015, "The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies," npj Computational Materials, 1(1), pp. 1-15.
[40] Wang, L., Chan, Y.-C., Ahmed, F., Liu, Z., Zhu, P., and Chen, W., 2020, "Deep generative modeling for mechanistic-based learning and design of metamaterial systems," Computer Methods in Applied Mechanics and Engineering, 372, p. 113377.
[41] Zhu, B., Zhang, X., Zhang, H., Liang, J., Zang, H., Li, H., and Wang, R., 2020, "Design of compliant mechanisms using continuum topology optimization: a review," Mechanism and Machine Theory, 143, p. 103622.
[42] Svanberg, K., 1987, "The method of moving asymptotes—a new method for structural optimization," International journal for numerical methods in engineering, 24(2), pp. 359-373.

