On Latent Variable Estimation using Derivative Gaussian Processes

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Soham Mukherjee

aus Kolkatta/Indien

Tübingen

2025

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	11.03.2026
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter/-in:	Prof. Dr. Paul-Christian Bürkner
2. Berichterstatter/-in:	Prof. Dr. Manfred Claassen
3. Berichterstatter/-in	Prof. Dr. Philipp Hennig

Summary

Latent variable modeling is a major field in statistical inference. Accurate estimation of latent variables requires a principled statistical approach. In this thesis, we develop a class of latent variable models using derivative Gaussian processes. Our methods include several extensions of Gaussian processes (GPs) namely latent variable GPs, multi-output GPs as well as derivative GPs under a single framework. We achieve this through a modified derivative covariance function that can handle multi-dimensional output data along with their derivatives to estimate latent variable inputs. Moreover, our models account for complexities in the underlying data such as scale differences between outputs and their derivatives, varying information across multiple outputs as well as interactions between outputs. Using Bayesian inference, our models provide uncertainty estimates for each latent variable sample. Through diverse simulation scenarios, we demonstrate that latent variable estimation accuracy can be significantly increased by including derivative information through our exact GPs. Additionally, we find that including derivatives without our proposed covariance structure yields misleading results, thus emphasizing the importance of our methods. Exact GPs, however, have limited applicability for larger datasets due to their steep computational complexity. The scalability issues are further aggravated upon combining all the aforementioned GP extensions. To overcome this, we extend the recently developed Hilbert space approximations initially for multi-output and latent input settings. Under the Hilbert space Gaussian process (HSGP) framework, the covariance function is approximated with a reduced-rank representation through its spectral decomposition computed from a finite set of basis functions. By exploiting the spectral representation of a stationary covariance function, HSGPs scale linearly with both sample size and the number of basis functions. Through various experiments, we show that HSGPs provide better posterior uncertainty calibration and estimation accuracy for latent variable samples. Compared to other GP approximations, our methods find a nice balance between trustworthy inference and speed when it comes to latent variable estimation. In case of the derivative GPs, their covariance structure is jointly defined. As a general case, we extend the Hilbert space methods for *composite GPs*, where we model a pair of data source as different outputs and obtain a spectral approximation of the composite covariance functions. As a special case of composite GPs, we then develop scalable derivative GPs by modeling the outputs along with their derivatives. Specifically, we derive and analyze the spectral decomposition of our modified derivative covariance functions and further study their properties theoretically. Through our extended Hilbert space approximations, the class of latent variable derivative GPs can be widely applicable in large sample data scenarios. As a concrete application, we showcase our methods on the estimation of the unobserved cellular ordering in the field of single-cell biology.

Zusammenfassung

Die Modellierung latenter Variablen ist ein wichtiger Bereich der statistischen Inferenz. Die genaue Schätzung latenter Variablen erfordert einen fundierten statistischen Ansatz. In dieser Arbeit entwickeln wir eine Klasse von Modellen latenter Variablen unter Verwendung derivative Gaussian processes. Unsere Methoden umfassen mehrere Erweiterungen von Gaussian-Processes (GPs), nämlich latente Variablen-GPs, Multi-Output-GPs sowie derivative GPs in einem einzigen Modell. Dies erreichen wir durch eine modifizierte derivative-Kovarianzfunktion, die mehrdimensionale Daten zusammen mit ihren Ableitungen verarbeiten kann, um latente Variableneingaben zu schätzen. Darüber hinaus berücksichtigen unsere Modelle Komplexitäten in den zugrunde liegenden Daten, wie zum Beispiel Skalenunterschiede zwischen Ausgaben und ihren Ableitungen, unterschiedliche Informationen über mehrere Ausgaben hinweg sowie korrelative Wechselwirkungen zwischen den mehrdimensionalen Ausgaben. Unter Verwendung Bayes'scher Inferenz liefern unsere Modelle quantitative Schätzungen der Ungenauigkeit jeder geschätzten latenten Variable. Anhand verschiedener Simulationsszenarien zeigen wir, dass die Genauigkeit der Schätzung latenter Variablen durch die Einbeziehung von Ableitungsinformationen über unsere exakten GPs erheblich gesteigert werden kann. Darüber hinaus stellen wir fest, dass die Einbeziehung von Ableitungen ohne unsere vorgeschlagene Kovarianzstruktur zu irreführenden Ergebnissen führt, was die Relevanz unserer Methoden unterstreicht. Exakte GPs sind jedoch aufgrund ihrer hohen Rechenkomplexität nur begrenzt auf größere Datensätze anwendbar. Diese Skalierbarkeitsprobleme verschärfen sich noch weiter, wenn alle oben genannten GP-Erweiterungen kombiniert werden. Um dies zu überwinden, erweitern wir die kürzlich entwickelten Hilbert-Raum-Approximationen zunächst für Multi-Output- und latente Eingabeeinstellungen. Im Rahmen des Hilbert-space-Gaussian-Processes (HSGP) wird die Kovarianzfunktion durch eine spectral decomposition, die aus einer endlichen Menge von Basisfunktionen berechnet wird, mit einer reduced-rank approximation. Durch die Nutzung der spectral representation einer stationären Kovarianzfunktion skalieren HSGPs linear sowohl mit der Stichprobengröße als auch mit der Anzahl der Basisfunktionen. Durch verschiedene Experimente zeigen wir, dass HSGPs eine bessere Kalibrierung der posterioren Unsicherheit und eine höhere Schätzgenauigkeit für latente Variablenstichproben bieten. Im Vergleich zu anderen GP-Approximationsmethoden finden unsere Methoden ein gutes Gleichgewicht zwischen zuverlässiger Inferenz und Geschwindigkeit, wenn es um die Schätzung latenter Variablen geht. Im Falle der derivative-GPs wird ihre Kovarianzstruktur gemeinsam definiert. Als allgemeinen Fall erweitern wir die Hilbert-space-Methoden für die *composite GPs*, bei denen wir ein Datenpaar gemeinsam als unterschiedliche Ausgänge modellieren und eine spektrale Approximation der composite Kovarianzfunktionen erhalten. Als Sonderfall der composite GPs entwickeln wir dann skalierbare derivative GPs, indem wir die Ausgänge zusammen mit ihren Ableitungen gemeinsam modellieren. Insbesondere leiten wir die spektrale Zerlegung von abgeleiteten Kovarianzfunktionen ab, analysieren sie und untersuchen ihre Eigenschaften weiter theoretisch. Durch unsere erweiterten Hilbert-Raum-Approximationen kann die Klasse der latenten variablen abgeleiteten GPs in Szenarien mit großen Datenmengen angewendet werden. Als konkrete Anwendung präsentieren wir unsere Methoden zur Schätzung der nicht beobachtbaren zellulären Ordnung von Transkriptionsdaten einzelner Zellen.

Acknowledgment

I thank my advisors Paul and Manfred for their supervision, support and crucial contributions to my research. During my PhD, I was part of a computational statistics group and an ML research group focused on applications in single-cell biology. I appreciate them for providing me this opportunity of working in this unique arrangement. I am grateful for all the things I have learned from them over these years.

I thank my PhD advisory committee member Philipp for his service and interest in my research. I thank IMPRS-IS for supporting my PhD by providing numerous training and networking opportunities. I thank all the IMPRS-IS coordination office members for making this possible.

I would like to take a moment and appreciate Matthias, Jan, Javier, Luna and Florence for their unconditional support throughout my PhD. You all played a major role in making it easy for me to thrive in an unknown land far away from home and enriching this whole experience. Thank you Matthias for being the friend you are and teaching me how the smallest acts of kindness goes a long way.

I am grateful to all my colleagues and friends. Here's an incomplete list of people who made the groups a great place to be over the years (apologies for missing anyone): Aayush, Daniel, Šimon, Svenja, Lars, Philipp, Max, Gabriel, Wenjie, Pouria, Vivek, Sepideh, Nicola, Rosanna, Anke.

Thank you Revant for all the academic as well as life related discussions we had on the white-board and the office balcony. Thank you Marcello for teaching me how to supervise at your own expense. Thank you Sebastian for being a great collaborator and showing how to never let go without understanding the topic at hand.

I extend my thanks to Rachel, Michael, Ana, Nacho for all the meetups and cooking sessions together. Thank you for constantly keeping in touch even after I moved to a different city. You were so much better at this than me. Thank you Colin, my first flatmate here, for helping me survive those first days as well as to move cities in between my PhD.

My heartiest gratitude goes to my long time school friends Debapratim, Pritam, Shourya and Tanwisha for tolerating me all these years. The numerous game evenings were just an excuse to get together even though we are scattered around the globe.

Lastly, I would take this opportunity to express my appreciation to my parents, grandparents and sister. Thank you all, especially my mother Jayati Chatterjee, for the important life lessons you instilled in me helping to build a strong foundation that led me here today. I owe my initial training and love for mathematics and statistics to my grandfather Nani Gopal Mukherjee. Specifically, I would thank my grandfather the Late Dr. Ramram Chatterjee and my father Dr. Somnath Mukherjee for being my academic inspirations. I would like to apologize for those moments of absence when you all needed me. No amount of gratitude is enough for what you all contributed in shaping my life.

Preface

The core of this thesis consists of three chapters based on the following publications and preprints.

Chapter 3

DGP-LVM: Derivative Gaussian process latent variable models

Soham Mukherjee, Manfred Claassen and Paul-Christian Bürkner

Stat Comput 35, 120 – Published 8 June 2025

https://doi.org/10.1007/s11222-025-10644-4

Chapter 4

Hilbert space methods for approximating multi-output latent variable Gaussian processes

Soham Mukherjee, Manfred Claassen and Paul-Christian Bürkner

arXiv preprint arXiv:2505.16919 – in review (2025)

https://doi.org/10.48550/arXiv.2505.16919

Chapter 5

Latent variable estimation with composite Hilbert space Gaussian processes

Soham Mukherjee, Javier Enrique Aguilar, Marcello Zago, Manfred Claassen and Paul-Christian Bürkner

arXiv preprint arXiv:2510.25371 – in review (2025)

https://doi.org/10.48550/arXiv.2510.25371

During my PhD studies, I contributed to the following preprint which is not part of the thesis

velotest: Statistical assessment of RNA velocity embeddings reveals quality differences for reliable trajectory visualizations

Sebastian Bischoff, Pavlin G. Poličar, **Soham Mukherjee**, Jakob H. Macke, Manfred Claassen, Cornelius Schröder

bioRxiv 2025.10.26.683064 – in review (2025)

https://doi.org/10.1101/2025.10.26.683064

Notations

Throughout the thesis, unbolded x represents a single number, boldface $\mathbf{x}$ represents a vector, and capital boldface $\mathbf{X}$ represents a matrix. An individual element of a vector is denoted with a subscript and without boldface. For example, the i^{th} element of a vector $\mathbf{x}$ is x_i. Functions $f(x)$ are written without boldface, however, vector of functions are represented as $\mathbf{f}$. Derivatives of a^{th} order are represented as $f^{(a)}$. The a^{th} and b^{th} order partial derivatives of functions with respect to their arguments are represented as $f^{(a,b)}$.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Latent variable modeling is a cornerstone of statistical inference [53]. Such models attempt to explain complex relationships between several manifest (or observed) variables through a latent (or an unobserved) variable. A standard approach is to assume a functional relationship between the latent variables and the response. This functional relationship encodes the dependency between the response and latent variables. Early works on that front involving latent variable models are in factor analysis [36] where the observed response variable depends on latent factors that are to be estimated. Later, latent factor analysis was unified with regression models under an umbrella term of structural equation modeling [45] where linear relationships between latent variables and the response variable are assumed. Other uses of latent variable models have been in classification [51] including dimension-reduction methods [49, 86]. In this thesis, we focus on modeling latent variables using a generalized functional regression approach through Gaussian processes (GPs) [63, 94].

GPs are a popular class of non-parametric probabilistic methods that are well known for their flexible modeling approaches. Based on their flexible structure, several extensions have been proposed so far. Among them, the relevant ones to this thesis are latent variable GPs [49, 50], multi-output GPs [25, 34, 44] and derivative GPs [77]. By design, latent variable GPs are intended to estimate latent variables. This is done by designating the inputs of the GPs as latent. While the earlier uses of these latent variable GPs were closer to factor analysis models, recently they have been developed as latent variable GP regressions [11]. In this approach, latent variables were introduced as an extension of standard GP regression models to model multi-modal non-stationary stochastic processes. In contrast, we directly assign the latent variable as a single-input dimension that is to be estimated given the output data. Since our primary objective is to accurately estimate the latent inputs along with their uncertainty estimates, we use a full Bayesian inference to obtain posteriors of latent variables via Markov Chain Monte Carlo (MCMC) sampling. Under this latent variable regression structure, each output observation has a corresponding latent input. Thus, the standard way of increasing sample size to increase accuracy in estimating our latent inputs remains infeasible. To alleviate this problem, we incorporate derivative information in addition to the observed outputs and multi-dimensional outputs in the same framework.

GPs allow jointly modeling two or more stochastic processes together as additional sources of information [69]. We refer to this general structure of modeling two different processes as *composite GPs*. When one of these outputs is a derivative of another, we have the special case of derivative GPs. Briefly, since differentiation is a linear operation, derivative of a GP is another GP [69]. Thus, a GP and its derivative can be jointly modeled together using a joint covariance structure involving, among others, the covariances of these two GPs [77]. Derivative GPs have not been studied in too much detail since their inception in [77]. Recently, they have been used in scaling GP regressions to large datasets [27, 65]. In contrast, we use the derivative information along with the original output so that the latent inputs are shared among them as a way of effectively increasing the amount of available information. From a practical perspective, the derivatives can be on a vastly different scale than the original GPs when they are not exactly computed from the original GPs. As a result, the derivatives would be only proportional to exact derivatives. Under such situations, standard GP specifications struggle to overcome the scale difference between the original GPs and their (proportional) derivatives under a single model framework. We overcome these issues through a custom joint covariance function for the derivative GP model.

To further increase the amount of information on the latent inputs, we consider the output to be matrix-variate having multiple output dimensions. In such cases, multi-output GPs simultaneously model these outputs as multiple tasks [12, 85]. Multi-output GPs model the shared information between different output dimensions via an across-dimension correlation matrix and are known to have improved performance in model inference as compared to modeling the uni-dimensional outputs separately [4, 6, 12]. Since each of the multiple outputs can carry specific information in real-world scenarios, we modify our covariance function to assign each of the outputs their own set covariance function hyperparameters. This way, we also account for the unique information carried by the multiple outputs. Under our framework, we have the latent inputs shared across all the outputs. This way, we pool the information from each output dimensions together, in addition to their derivatives, to increase the estimation accuracy of latent inputs.

Exact GPs have a practical limitation when it comes to data scenarios involving large samples [69]. This is due to the requirement of solving the Gram matrix generated from the covariance function used to specify the GP, resulting in a cubic computational complexity with respect to sample size. When the aforementioned extensions are considered in a single model, the scalability issues are aggravated to the point that any data scenarios involving more than a few hundreds of sample sizes are impractical. This happens since adding derivative information already doubles the sample size. Further, considering multiple output dimensions increase the computational complexity due to the requirement of additionally solving the across-output dimension correlation matrix modeling the shared information between outputs. To overcome these scalability issues, several methods have been proposed for approximating the covariance functions through a reduced-rank representation [69, 73, 75, 76]. These methods are based on using inducing points [67] along with approximate model inference via mean-field variation inference [87]. With these approximations, GPs scale linearly with sample size and quadratically

with the number of inducing points. However, choosing the representative inducing points for the specific latent inputs structure we have is ambiguous. Moreover, an in-depth study of the statistical properties of latent variable estimates using these methods is yet to be done.

An alternative reduced-rank approach is to approximate the covariance function through its spectral decomposition computed from a finite set of basis functions. This method falls under the category of the recently developed Hilbert space approximate GPs (HSGPs) [72, 78]. By exploiting the spectral representation of stationary covariance functions, the computational complexity of HSGPs scales linearly with both sample size and the number of basis functions. However, HSGPs have so far only been developed for single outputs and manifest (known) inputs. Thus, in this thesis, we develop extensions for the HSGP framework to address latent variable inputs, multi-dimensional outputs as well as derivative GPs. Inference for HSGPs are typically carried out via MCMC sampling, which can lead to slower inference compared to other GP approximations but produces posterior approximations of much higher quality [97]. We show an in-depth study of the statistical properties and benefits of HSGPs over exact GPs as well as other GP approximation methods.

Generalizing Hilbert space methods for derivative GPs goes beyond merely obtaining the spectral densities of the covariance functions. Due to the joint covariance structure of derivative GPs, the functional forms involved need to satisfy the requirement of the Hilbert space methods. Thus, we obtain the functional forms of spectral densities for the general derivatives of a chosen stationary covariance function along with theoretical results on conditions under which they are indeed valid covariance functions. As a remedy to the joint derivative covariances, we develop the spectral decomposition of a partial structure involving the GPs and their derivative process. Using a block-wise treatment of the general joint covariance structure, we show that the Hilbert space methods can be extended resulting in scalable class of derivative GPs, and more generally composite GPs. To determine the trade-off in modifying the joint covariance structure and their approximations, we compare them to the our exact derivative GPs along with other competitive models in terms of inference speed and latent variable estimation accuracy.

As an illustration, we apply our methods on real-world case studies for estimating cellular ordering based on gene expression levels [90], a contemporary research problem in single-cell biology. Through our developed methods, we tackle the problem of analyzing various sources of gene expression levels as well as their derivative information RNA velocity [47]. By doing so, we attempt to accurately estimate latent cellular ordering, and increase our understanding of the underlying biological process.

1.1 Outline of the thesis

The rest of the thesis is designed as follows.

Chapter 2 outlines the fundamental concepts involved in GPs and their extensions. We discuss the relevant properties of the covariance functions, their choices as well as derivative forms. We mention the preliminary details of the Hilbert space approximations and briefly discuss

the inducing points methods as relevant approximation strategies. Lastly, we discuss inference strategies involved for our developed methods before providing details on an single-cell biology application to illustrate our methods.

In Chapter 3, we develop an exact derivative latent variable GP that models multi-dimensional output data using our modified covariance functions. We demonstrate how we increase latent variable estimation accuracy when we use our proposed GP modifications in a single model framework. We validate our methods under diverse simulation scenarios and showcase the benefits of our developed methods.

In Chapter 4, we overcome the scalability limitations of multi-output latent variable GPs using our extensions to the Hilbert space approximation method. We demonstrate the superior model inference speed compared to their exact model counterparts. Moreover, we compare our approximations to other GP approximation methods in terms of accuracy and uncertainty calibrations for latent variable estimation.

Chapter 5 further extends the Hilbert space approximations for composite GPs along with a specific focus on derivative GPs. We obtain theoretical results and conditions under which the Hilbert space methods can be used to approximate derivative covariance functions. We show how we overcome the practical limitations of our exact model from Chapter 3 and develop a scalable method for latent variable estimation using derivative GPs before illustrating our methods on real-world case studies.

Finally, in Chapter 6, we summarize our contributions before outlining the limitations and future work for derivative latent variable GPs and their applications.

Chapter 2

Background

In this chapter, we touch upon the fundamental concepts of the topics we develop in the rest of the thesis. We briefly discuss the basic structure of GPs and covariance functions along with their properties that we later use. Following that, we mention the different extensions of GPs before discussing approximation strategies. Lastly, we present the concepts involved in our biological applications.

2.1 Gaussian processes

Gaussian processes or GPs are a general class of probabilistic models for functions that were introduced in [63] as a non-linear regression method. Consider $\mathbf{x} = \{x_1, \ldots, x_N\}$ to be a set of input variables for N sample size. Then $\mathbf{f}$ is a GP if any finite set of functions $\{f(x_1), \ldots, f(x_N)\}$ follows a joint Gaussian distribution [69]. GPs are specified as $f(x) \sim \mathcal{GP}(m(x), k(x, x'))$ where $m(x)$ is a mean function and $k(x, x')$ is a covariance function for $x, x' \in \mathbb{R}$. The mean and covariance functions completely specify the GPs $\mathbf{f}$. Specifically, the choice of covariance functions determine the behavior of the functions and how they generalize to modeling data.

To model data $\mathbf{y}$, we consider a pair of variables $(\mathbf{y}, \mathbf{x})$ where $\mathbf{y}$ is an univariate output (response) and $\mathbf{x}$ is the input (covariate) with observations $y_i, x_i \in \mathbb{R}$, $i = \{1, \ldots, N\}$. Then, the relationship of $\mathbf{y}$ and $\mathbf{x}$ using a GP $\mathbf{f}$ is written as

$$y_i = f(x_i) + \varepsilon_i \tag{2.1}$$

where ε_i is the i^{th} sample of additive noise $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Eq.(2.1) can be equivalently expressed as the likelihood $p(\mathbf{y} \mid \mathbf{f}) \sim \mathcal{N}(f(\mathbf{x}), \sigma^2)$. Thus, for $i \neq j$, $\mathrm{Cov}(y_i, y_j) = k(x_i, x_j)$ and for $i = j$, we have $\mathrm{Var}(y_i) = k(x_i, x_j) + \sigma^2$. Together, we obtain the Gram matrix $\mathbf{K}$ where $k(x_i, x_j)$, $i \neq j$ are the $(i, j)^{th}$ element of $\mathbf{K}$ with $k(x_i, x_i)$ being the diagonals. We will discuss examples of covariance functions in Section 2.2.2.

2.1.1 Predictions on Gaussian processes

A central property of GPs is the availability of a closed form predictive distribution [69, 94]. Given the GPs $\mathbf{f}$, the joint distribution for another set of GP functions $\mathbf{f}_*$ is

$$\begin{bmatrix}\mathbf{f}\\ \mathbf{f}_*\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{\mu}_f\\ \boldsymbol{\mu}_{f_*}\end{bmatrix}, \begin{bmatrix}\mathbf{K}_{ff} & \mathbf{K}_{ff_*}\\ \mathbf{K}_{f_*f} & \mathbf{K}_{f_*f_*},\end{bmatrix}\right) \tag{2.2}$$

such that $\mathbf{K}_f$ and $\mathbf{K}_{f_*f_*}$ are the Gram matrices obtained from the covariance functions of $\mathbf{f}$ and $\mathbf{f}_*$. $\mathbf{K}_{ff_*}$ and $\mathbf{K}_{f_*f}$ are matrices denoting the interactions between $\mathbf{f}$ and $\mathbf{f}_*$. The means $\boldsymbol{\mu}_f$ and $\boldsymbol{\mu}_{f_*}$ are obtained via the mean functions of the corresponding GPs.

Combining the joint prior $p(\mathbf{f}, \mathbf{f}_*)$ with the likelihood $p(\mathbf{y} \mid \mathbf{f})$ and using Bayes rule, we obtain the joint posterior

$$p(\mathbf{f}, \mathbf{f}_* \mid \mathbf{y}) = \frac{p(\mathbf{f}, \mathbf{f}_*)p(\mathbf{y} \mid \mathbf{f})}{p(\mathbf{y})}. \tag{2.3}$$

The posterior predictive distribution $p(\mathbf{f}_* \mid \mathbf{y})$ is then obtained by marginalizing out $\mathbf{f}$. Since both $p(\mathbf{f}, \mathbf{f}_*)$ and $p(\mathbf{y} \mid \mathbf{f})$ are Gaussian, the predictive distribution has a closed form

$$p(\mathbf{f}_* \mid \mathbf{y}) = \mathcal{N}(\mathbf{K}_{f_*f}(\mathbf{K}_{ff} + \sigma^2\mathbf{I})^{-1}\mathbf{y}, \mathbf{K}_{f_*f_*} - \mathbf{K}_{f_*f}(\mathbf{K}_{ff} + \sigma^2\mathbf{I})^{-1}\mathbf{K}_{ff_*}), \tag{2.4}$$

where the $\sigma^2\mathbf{I}$ is matrix of the error variances based on ε in Eq.(2.1). While we do not directly use the predictive distribution in this thesis, we eventually use this joint GP prior structure in Eq.(2.2) to obtain *composite GPs*. We discuss the composite GPs and their demonstrate their developments in Chapter 5.

2.2 Covariance functions

A covariance function k (also called a covariance kernel) is a positive definite, symmetric function of two inputs x, $x' \in \mathbb{R}$ [31]. A Gram matrix $\mathbf{K}$ generated using a covariance function $k(x, x')$ is called a covariance matrix [69]. For a GP $f(x)$, the covariance function k is used to denote the prior covariance between two arbitrary function values such that

$$\mathrm{Cov}(f(x_i), f(x_j)) = k(x_i, x_j) \quad i, j = \{1, \ldots, N\} \tag{2.5}$$

The covariance function k is symmetric if $k(x, x') = k(x', x)$. k is positive definite if it satisfies

$$\int k(x, x')f(x)f(x')d\nu(x)d\nu(x') > 0 \tag{2.6}$$

for all $f \in L_2(\mathcal{X}, \nu)$ where $\mathcal{X}$ denotes the input space with a measure ν. The corresponding Gram matrix $\mathbf{K}$ is then symmetric and positive definite having positive eigenvalues. Based on these conditions, we can determine or construct a valid covariance function [31, 57].

A covariance function k is stationary if it is a function of $r = x - x'$. Further, it is isotropic if $r = \mid x - x' \mid$. A stationary covariance function can be represented using a positive finite spectral density as its Fourier dual. We will state two theorems that are central to this idea: Bochner's

theorem [31, 69, 82] and Wiener-Khintchine theorem [23, 93].

Theorem 2.1 (Bochner's theorem)**.** *A complex-valued function k on $\mathbb{R}$ is the covariance function of a mean square continuous complex valued random process on $\mathbb{R}$ if and only if it can be represented as*

$$k(r) = \int_{\mathbb{R}} \exp(2\pi i\omega r)\, dS(\omega) \tag{2.7}$$

where S is a positive finite measure and ω represents the inputs in the frequency domain.

We will expand more on the mean square continuity as well as differentiability ideas below in Section 2.2.1.

Theorem 2.2 (Wiener-Khintchine theorem)**.** *For any real-valued stationary stochastic process with covariance function $k(r)$, there exists a density function $S(\omega)$ such that*

$$S(\omega) = \int_{\mathbb{R}} k(r)\, \exp(-i\omega r)\, dr \tag{2.8}$$

and $k(r)$ and $S(\omega)$ are Fourier transforms of each other.

We will revisit these two theorems in Chapters 4 and 5 where we develop our approximation methods using them as building blocks.

2.2.1 Differentiability of covariance functions

We will outline the mean square continuity and differentiability concepts for stochastic processes briefly. For a sequence of random variables $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, let a random variable $\mathbf{x}_*$ defined on some probability space such that $\mid \mathbf{x}_N - \mathbf{x}_* \mid \to 0$ as $N \to \infty$ pointwise. Then, a process $f(\mathbf{x})$ is continuous in the mean square sense if $\mathbb{E}(\mid f(\mathbf{x}_N - f(\mathbf{x}_*) \mid^2) \to 0$ as $N \to \infty$ [69, 82]. A random field is mean square continuous at $\mathbf{x}_*$ if and only if its covariance function $k(\mathbf{x}, \mathbf{x}')$ is continuous at the point $\mathbf{x} = \mathbf{x}' = \mathbf{x}_*$. For stationary covariance functions this reduces to just checking continuity at $k(0)$.

The mean square differentiability of $f(\mathbf{x})$ denoted by $f^{(1)}(\mathbf{x})$ with respect to a sample x_i is defined as [2, 69]

$$f^{(1)}(\mathbf{x}) = \frac{\delta f(\mathbf{x})}{\delta x_i} = \lim_{h \to 0} \frac{f(\mathbf{x} + he_i) - f(\mathbf{x})}{h} \tag{2.9}$$

when the limit exists in the mean square sense with e_i as the unit vector in the i^{th} direction. The covariance function for $f^{(1)}(\mathbf{x})$ is given by $k^{(1,1)}(\mathbf{x}, \mathbf{x}') = \delta^2 k(\mathbf{x}, \mathbf{x}')/\delta x_i \delta x'_j$. For a GP, if the $2a^{th}$ partial derivative $k^{(a,a)}(\mathbf{x}, \mathbf{x}')$ exists for $a \geq 1$, $a \in \mathbb{N}$, then the a^{th} order derivative process $f^{(a)}(\mathbf{x})$ exists. This result is used throughout this thesis when it comes to constructing derivative GPs. We further discuss the theoretical results for the general order differentiability and their conditions in Chapter 5 along with the proofs in Appendix C.

2.2.2 Choice of covariance functions

GPs are specified primarily by choosing a covariance function. Among the many choices of available covariance functions, the most popular choice is perhaps the Squared Exponential (SE) covariance function. This covariance function has a length-scale parameter $\rho > 0$ and a

standard deviation (SD) parameter $\alpha > 0$ also called the GP marginal SD. The length-scale ρ dictates the distance between two arbitrary points in the input space, beyond which they are uncorrelated. In that sense, ρ is dependent on the input space. The GP marginal SD, as the name suggests, denotes the spread of the GP functions. The SE covariance function for inputs x, x' is given by

$$k(x, x') = \alpha^2 \exp\left(-\frac{|\, x - x' \,|^2}{2\rho}\right). \tag{2.10}$$

It is straightforward to see that the SE covariance function is stationary and isotropic by setting $r =| \, x - x' \, |$. Additionally, the SE covariance function has mean square derivatives of all orders, that is, $k^{(a,a)}(x, x')$ exists for any $a \geq 1$, $a \in \mathbb{N}$. Thus, using a SE covariance function, it is possible to obtain derivative of GPs denoted by $f^{(a)}(x)$ of all orders. We prefer using the SE covariance function in this thesis primarily for the infinitely differentiable property. However, the infinite differentiable property renders the function to have strong smoothness, which might be unrealistic for modeling many real-world scenarios [82]. This can be overcome by using the more general Matérn family of covariance functions. The Matérn family is also specified using the length-scale ρ and GP marginal SD α. Additionally, the general form uses a scale parameter $\nu > 0$ that dictates the smoothness of the GP functions (higher ν results in more smooth functions). The Matérn family of covariance functions with argument $r =| \, x - x' \, |$ is given as

$$k(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu r}}{\rho}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu r}}{\rho}\right), \tag{2.11}$$

where K_ν is a modified Bessel function [1]. If $\nu \to \infty$, we obtain the SE covariance function. Depending on ν, the Matérn family members have a finite order of differentiability. Specifically, if Matérn covariance functions are expressed in the form of $\nu = p + 1/2$, where p is a non-negative integer, the Matérn is also p times mean square differentiable [69]. We discuss the specific derivative forms of Matérn family members and SE covariance function in Chapter 3 and Appendix A with theoretical results pertaining to the differentiability conditions and their spectral densities in Appendix C.

2.3 Extensions of Gaussian processes

In this section, we present a brief overview of the various extensions of GPs that are relevant to our methods in this thesis but discuss them in-depth in the following chapters. Namely, these extensions are latent variable GPs, multi-output GPs and derivative GPs.

Latent variable GPs, as the name suggests designates the input space to be latent as opposed to observed (in standard GP models). The latent variable GPs have been originally used for dimension-reduction methods and classification tasks [49, 50]. Later, the latent variable estimation was incorporated under the GP regression framework [11]. An unifying work on latent variable GPs was presented recently in [48]. The latent variable GPs are considerably more challenging than their manifest counterparts. This is due to several identifiability issues stemming from estimating the length scale parameter which is dependent on the inputs that are now latent (and are to be estimated as well). In this thesis, we show how to impose

an informed prior structure on the latent inputs and overcome this identifiability issue using Bayesian inference via MCMC sampling. The exact treatment of latent variables are presented in the later chapters.

When we want to model multiple output variables $(\mathbf{y}_1, \ldots, \mathbf{y}_D)$ (with D being the number of outputs) simultaneously as opposed to a single output $\mathbf{y}$, we consider multi-output GPs. The aim here is to model the shared information between each outputs which leads to increased performance in model inference as compared to modeling the uni-dimensional outputs separately [12]. This is achieved by modeling each output dimensions with an individual GP and then linearly combine them using a correlation matrix [12, 85]. This method of simultaneous modeling a matrix-variate response variable is called the *linear model of coregionalization* [25, 34, 44]. Under this method, the GP has a two-fold covariance structure: one between output samples and one across output dimensions. The former is treated similar to any standard GPs using a covariance function. In the simplest scenarios, the across outputs correlation is modeled using a uniform correlation matrix that assigns weights to the corresponding GPs before linearly combining them. Specifically, for an across outputs correlation matrix $\mathbf{C}$ and GPs $(\mathbf{f}_1, \ldots, \mathbf{f}_D)$, each i^{th} sample is obtained as

$$(f_1^*(x_i) \ldots f_D^*(x_i))^T = \mathbf{A} \times (f_1(x_i) \ldots f_D(x_i))^T \tag{2.12}$$

where $(.)^T$ denotes the transpose, $\mathbf{A}$ is the Cholesky factor of $\mathbf{C}$ such that $\mathbf{C} = \mathbf{A}\mathbf{A}^T$. Alternatively, one can specify $\mathbf{C}$ using input-dependent covariance functions [13, 32] however that requires application specific knowledge. In this thesis, we only consider the simpler case of uniform across outputs correlations.

We introduced the structure of derivative of GPs earlier in Section 2.2.1. To jointly model a standard GP $\mathbf{f}$ and its derivative $\mathbf{f}^{(1)}$, we use the derivative GPs [77]. The derivative GPs have a special joint derivative covariance function structure that consists of the covariance functions for both the GPs and the functions that model their interactions. The specific derivative GP structure we consider is given in Chapter 3. The use of derivative observations were previously discussed in [64]. In our case, we use the derivative information such that the latent inputs are shared between $\mathbf{f}$ and $\mathbf{f}^{(1)}$. This way, we increase the pool of available information to better estimate the latent inputs. However, if there are considerable scale differences between outputs and their derivatives, standard GPs struggle to fit both these quantities under a single framework. We overcome these issues through our proposed derivative GP structure through a custom covariance function. The details of our theoretical developments for derivative GPs are provided in Chapter 3 and subsequently in Appendix A. Further, the theoretical results involving existence of the general derivative GPs are shown in Chapter 5 and Appendix C.

2.4 Approximating Gaussian processes

While GPs have favorable theoretical properties, they are often limited in applications due to their computational complexity. Fitting a GP involves solving the Gram matrix $\mathbf{K}$ of dimensions $N \times N$ corresponding to the covariance function k with N being sample size. Thus, for large

N, such computations are not practically feasible. To overcome these limitations, one approach is to use a reduced-rank representation of $\mathbf{K}$ which involves approximating the Gram matrix with a lower rank ($< N$) matrix. We will discuss two major approaches that fall under this approximation category.

2.4.1 Spectral approximations

The optimal reduced-rank approximation $\mathbf{K}$ with respect to the Frobenius norm is $\tilde{\mathbf{K}} = \mathbf{\Phi}\mathbf{\Delta}\mathbf{\Phi}^T$ where $\mathbf{\Delta}$ is a diagonal matrix of the leading M eigenvalues of $\mathbf{K}$ and $\mathbf{\Phi}$ is the matrix of the corresponding eigenvectors [33, 69]. However, directly computing this eigen-decomposition is an $O(N^3)$ operation and thus similarly prohibitive. Recently, [78] developed a method for obtaining approximate eigenvalue decompositions of covariance functions using an eigenfunction expansion of the Laplace operator in a compact subset of $\mathbb{R}$. This method utilizes the Hilbert space methods [24] to interpret the approximate covariance function through its spectral representations.

Consider inputs $x, x' \in \mathbb{R}$ and ω as their representations in the frequency domain. Using Theorems 2.1 and 2.2, we obtain a spectral density $S(\omega)$ for a stationary isotropic covariance function $k(r)$ where $r =\mid x - x' \mid$. By defining a covariance operator $\mathcal{K}$, we write

$$\mathcal{K}\phi = \int k(., x')\phi(x')dx' \tag{2.13}$$

where $\phi(.)$ is the eigen function corresponding to $\mathbf{K}$ generated from k [78]. Using the isotropic property, we write $S(\omega) \triangleq S(||\omega||)$ where $||.||$ denotes the Euclidean norm. Assuming an analytic functional form for the spectral density $S(\omega)$, it can be written as a polynomial expansion

$$S(||\omega||) = a_0 + a_1||\omega|| + a_2(||\omega||^2)^2 + a_3(||\omega||^2)^3 + \dots \tag{2.14}$$

Since the transfer function corresponding to Laplace operator $\nabla^2 = -||\omega||^2$ [78], we take the inverse Fourier transform of Eq.(2.14) to obtain the representation

$$\mathcal{K} = a_0 + a_1(-\nabla^2) + a_2(-\nabla^2)^2 + a_3(-\nabla^2)^3 + \dots \tag{2.15}$$

One way to obtain the Hilbert space approximation for Eq.(2.15) is through Dirichlet boundary conditions on a compact subset $\Omega \subset \mathbb{R}$ as

$$\begin{aligned} -\nabla^2\phi_j(x) &= \lambda_j\phi_j(x), \quad x \in \Omega \\ \phi_j(x) &= 0, \quad x \notin \Omega \end{aligned} \tag{2.16}$$

With $-\nabla^2$ being a positive definite Hermitian operator, the eigenfunctions ϕ_j is orthonormal with respect to its inner product and eigenvalues λ_j are real and positive [78]. Using a formal kernel on the negative Laplace operator $l(x, x') = \sum_j \lambda_j\phi_j(x)\phi_j(x')$ such that $-\nabla^2 g(x) = \int l(x, x')g(x')dx'$ for any differentiable function g in the domain Ω. The orthonormality of the basis allows $l^s(x, x') = \sum_j \lambda_j^s\phi_j(x)\phi_j(x')$ and subsequently $(-\nabla^2)^s g(x) = \int l^s(x, x')g(x')\, dx'$ for

arbitrary power $x = 1, 2, \ldots$. Considering the boundary conditions from Eq.(2.16), we have

$$\begin{aligned}&[a_0 + a_1(-\nabla^2) + a_2(-\nabla^2)^2 + a_3(-\nabla^2)^3 + \ldots]g(x) \\ &= \int [a_0 + a_1 l^1(x, x') + a_2 l^2(x, x') + a_3 l^3(x, x') + \ldots]g(x')\, dx'\end{aligned} \tag{2.17}$$

Comparing Eq.(2.17) with Eqs.(2.15) and (2.13), we obtain

$$\begin{aligned}k(x, x') &\approx a_0 + a_1 l^1(x, x') + a_2 l^2(x, x') + a_3 l^3(x, x') + \ldots \\ &= \sum_j [a_0 + a_1 \lambda_j^1 + a_2 \lambda_j^2 + a_3 \lambda_j^3 + \ldots]\phi(x)\phi(x')\end{aligned} \tag{2.18}$$

Further, setting $||\omega||^2 = \lambda_j$ we obtain the covariance function approximation with its spectral density

$$k(x, x') \approx \sum_j S(\sqrt{\lambda_j})\phi(x)\phi(x') \tag{2.19}$$

under the domain Ω with λ_j and ϕ_j as the j^{th} eigenvalue and eigenfunction respectively. This infinite sum in Eq.(2.19) can now be further approximated using a finite $M < N$ number of basis points and obtain a reduced-rank representation that results in a scalable approximation. Further details on applying this to speed up inference for GPs and their extensions are discussed in Chapters 4 and 5.

2.4.2 Inducing points

An older, and perhaps more popular approach to the reduced-rank representation of covariance functions are by inducing points methods [69, 73, 75, 76]. An unifying work on inducing points is presented in [67]. In this method, the covariance functions is approximated by introducing a set of M latent variables $\mathbf{u} = \{u_1, \ldots, u_M\}$. Although M has a fundamentally different implication here compared to the spectral approximations mentioned above in Section 2.4.1, we use the same notation for simplicity.

For a set of training and test GPs $\mathbf{f}$ and $\mathbf{f}_*$ respectively, these latent variables $\mathbf{u}$ denote the GP values and their corresponding set of inputs $\mathbf{x}_u$ are called inducing inputs. Involving the inducing variables $\mathbf{u}$ gives rise to the joint GP prior $p(\mathbf{f}, \mathbf{f}_*, \mathbf{u})$. However, using the consistency of GPs [67], we recover the joint GP prior $p(\mathbf{f}, \mathbf{f}_*)$ from Eq.(2.2) as

$$p(\mathbf{f}, \mathbf{f}_*) = \int p(\mathbf{f}, \mathbf{f}_*, \mathbf{u})\, d\mathbf{u} = \int p(\mathbf{f}_*, \mathbf{f} \mid \mathbf{u}) p(\mathbf{u})\, d\mathbf{u}, \tag{2.20}$$

where $\mathbf{u} \sim \mathcal{N}(0, \mathbf{K}_{u,u})$. The joint prior is approximated by assuming $\mathbf{f}$ and $\mathbf{f}_*$ to be conditionally independent given $\mathbf{u}$ such that

$$p(\mathbf{f}, \mathbf{f}_*) \simeq q(\mathbf{f}, \mathbf{f}_*) = \int q(\mathbf{f}_* \mid \mathbf{u}) q(\mathbf{f} \mid \mathbf{u}) p(\mathbf{u})\, d\mathbf{u}. \tag{2.21}$$

Under this assumption, $\mathbf{f}$ and $\mathbf{f}_*$ only communicate between each other through $\mathbf{u}$. In other words, $\mathbf{u}$ induces dependencies between $\mathbf{f}$ and $\mathbf{f}_*$, thus having the name inducing variables. The

exact expressions for the two conditionals $\mathbf{f} \mid \mathbf{u}$ and $\mathbf{f}_* \mid \mathbf{u}$ follows

$$\mathbf{f} \mid \mathbf{u} \sim \mathcal{N}(\mathbf{K}_{f,u}\mathbf{K}_{u,u}^{-1}\mathbf{u}, \mathbf{K}_{f,f} - \mathbf{Q}_{f,f}) \quad \text{and} \quad \mathbf{f}_* \mid \mathbf{u} \sim \mathcal{N}(\mathbf{K}_{f_*,u}\mathbf{K}_{u,u}^{-1}\mathbf{u}, \mathbf{K}_{f_*,f_*} - \mathbf{Q}_{f_*,f_*}), \quad (2.22)$$

where $\mathbf{Q}_{a,b} \triangleq \mathbf{K}_{a,u}\mathbf{K}_{u,u}^{-1}\mathbf{K}_{u,b}$. Under this specification, the task of solving $\mathbf{K}_{f,f}$ of $N \times N$ dimensions is reduced to solving $\mathbf{K}_{u,u}$ of dimensions $M \times M$ where $M < N$.

Various computational methods have been proposed to approximate these conditionals $\mathbf{f} \mid \mathbf{u}$ and $\mathbf{f}_* \mid \mathbf{u}$. Among them, the popular ones are the subset of regressors (SoR) [74, 75], deterministic training conditional (DTC) approximation [26, 73], fully independent training conditional (FITC) approximation [76] and variational free energy (VFE) approximation [87]. Later works recommend VFE approximation due to its superior properties over other inducing points methods [8]. The available VFE implementations both approximates the model using inducing points as well as the inference using mean-field variational inference [87]. A latent variable version of this approximation [88] is used as a competitive method to our developed spectral approximations, the details of which are presented in Chapter 4.

2.5 Inference

GPs naturally fit into the Bayesian paradigm due to the inherent *prior* specification on the functions $\mathbf{f}$. The inference for exact manifest GPs can be straightforward due to the presence of a closed form analytical solution. In contrast, latent variable GPs are challenging since the posterior for the latent inputs are intractable. Our primary objective is to estimate the latent inputs $\mathbf{x}$ along with the collection of GP parameters that can be expressed as $\boldsymbol{\theta}$ given the data $\mathbf{y}$. Since the accuracy and the quality of estimating the latent inputs, in this case, is of utmost importance, we prefer full Bayesian treatment using MCMC sampling [14, 98] for obtaining high quality posterior samples of $\mathbf{x}$ and $\boldsymbol{\theta}$. Thus, for latent variable $\mathbf{x}$ with the data $\mathbf{y}$, we can write the joint probability density $p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})$ after marginalizing out GP functions $\mathbf{f}$ as

$$p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}) = p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta})\, p(\mathbf{x})\, p(\boldsymbol{\theta}). \quad (2.23)$$

where $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta})$ denote the GP-based likelihood, $p(\boldsymbol{\theta})$ as the prior distribution on the GP parameters and $p(\mathbf{x})$ as the prior on the latent variable inputs. The GP parameters are assumed to have independent priors [89]. Taking the example of the SE covariance function with length-scale ρ, GP marginal SD α and error SD σ, the prior on GP parameters $\boldsymbol{\theta}$ can be written as

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta}) = p(\rho)\, p(\alpha)\, p(\sigma_d). \quad (2.24)$$

Using Bayes' rule, we then obtain the joint posterior over latent inputs $\mathbf{x}$ and GP parameters $\boldsymbol{\theta}$ as

$$p(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}) = \frac{p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})}{\int\int p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta})\, d\mathbf{x}\, d\boldsymbol{\theta}}. \quad (2.25)$$

Posterior samples of $\mathbf{x}$ and $\boldsymbol{\theta}$ are obtained via state-of-the-art adaptive Hamiltonian Monte Carlo [42, 62]. Including the multi-output and derivative information requires modifying the joint probability density from Eq.(2.23) and subsequently the joint posterior from Eq.(2.25).

The specific details of inference for our developed methods are discuss across the following respective chapters.

2.6 Application on single-cell biology

Biological processes in developmental biology requires stochastic approaches to understand cellular progression [55]. The primary question here is to determine how cells develop and undergo changes in their state throughout various stages over the period of time [54, 68]. This problem is tackled with gene expression levels obtained from single-cell RNA sequencing (scRNA-seq) technologies [39]. The measured gene expressions provide information about the cell states and how they changes over time. However, due to the limitations of the sequencing methods, each cell gets destroyed when their gene expressions are measured and are thus observed at a single time point. This induces discontinuity in the temporal sequence making it difficult to infer cell state transitions. A way to recover the continuous temporal development is to estimate the pseudotime ordering, a popular approach to describe such a biological process as temporal cellular sequence based on their states [90].

Current pseudotime ordering methods utilize these cell state snapshots to estimate pseudotime [3, 20]. However, a directional information about cell state changes has been accounted for in this task recently [37]. This directional information is obtained through RNA velocity which is estimated from the difference in unspliced and spliced gene expression levels over latent experimental time (different from pseudotime) [9, 47]. This RNA velocity can be assumed as estimated derivative of spliced gene expression data over time.

A standard single-cell RNA sequencing data consists of multiple genes per cell, unique information carried by each gene along with interactions among themselves. Modeling such data thus requires including multi-dimensional outputs with varying parameters accounting for gene-specific information as well as possible correlations among genes. Additionally, including RNA velocity simultaneously adds another layer of complexity where the model must include derivative information as well. Since RNA velocity is only an estimated derivative [35] of gene expression levels, they are on a significantly different scale than the gene expression levels. Dealing with this scale difference is a challenge that the model must overcome to provide reliable pseudotime estimates.

Due to the clustering mechanism of cells based on their types [39], the single-cell RNA sequencing data doesn't satisfy the independent samples assumption. Thus, a multivariate approach such as GPs become a natural choice for modeling such data. Using the multi-output, derivative and latent input extensions of GPs, we additionally account for all of the aforementioned complexities in analyzing single-cell RNA expression levels to estimate latent cellular ordering. Hence, we illustrate our methods on such real-world case studies from single-cell biology.

Chapter 3

DGP-LVM: Derivative Gaussian process latent variable models

We develop a framework for derivative Gaussian process latent variable models (DGP-LVMs) that can handle multi-dimensional output data using modified derivative covariance functions. The modifications account for complexities in the underlying data generating process such as scaled derivatives, varying information across multiple output dimensions as well as interactions between outputs. Further, our framework provides uncertainty estimates for each latent variable samples using Bayesian inference. Through extensive simulations, we demonstrate that latent variable estimation accuracy can be drastically increased by including derivative information due to our proposed covariance function modifications. The developments are motivated by a concrete biological research problem involving the estimation of the unobserved cellular ordering from single-cell RNA (scRNA) sequencing data for gene expression and its corresponding derivative information known as RNA velocity. Since the RNA velocity is only an estimate of the exact derivative information, the derivative covariance functions need to account for potential scale differences. In a real-world case study, we illustrate the application of DGP-LVMs to such scRNA sequencing data. While motivated by this biological problem, our framework is generally applicable to all kinds of latent variable estimation problems involving derivative information irrespective of the field of study.

Declaration

This chapter is based on the following published manuscript:

> *DGP-LVM: Derivative Gaussian process latent variable models*
> **Soham Mukherjee**, Manfred Claassen and Paul-Christian Bürkner
> Stat Comput 35, 120 – Published 8 June 2025
> https://doi.org/10.1007/s11222-025-10644-4

Text, figures and tables are adapted from the manuscript https://arxiv.org/abs/2404.04074 with minor updates in notations.

Author contributions

Soham Mukherjee: Conceptualization, Methodology, Implementation, Validation, Formal Analysis, Writing – Original Draft, Writing – Review and Editing.

Manfred Claassen: Conceptualization, Writing – Review and Editing, Supervision.

Paul-Christian Bürkner: Conceptualization, Methodology, Writing – Review and Editing, Supervision.

3.1 Introduction

Gaussian processes (GPs) are a class of statistical models known for their flexible structure and favorable properties to analyze complex data [95]. Since their inception, several extensions have been proposed, among which the most relevant ones to this paper are adding derivative information to GPs [69, 77], building GPs with multiple outputs [25, 69, 85] as well as modeling latent input variables [49, 50]. Since differentiation is a linear operation, any variable and its derivative would be linearly related. For the same reason, as a fundamental property of GPs, a derivative of a GP is just another GP with a related covariance function. Together, this results in a single GP model for the outputs and their derivatives with a joint derivative covariance function [77].

Derivative GPs have usually been studied and designed to model a single vector-valued output and have not been extended for multiple outputs. Multi-output GPs are most suitable when the response or output of the model contains multiple features, each expressed by its own dimension. One could fit individual GPs for each feature but then risks substantial loss of information in case of interactions between features. Thus a two-fold covariance structure was suggested in [85] allowing GPs to account for this shared information between features. We extend this two-fold covariance structure to derivative GPs. As we will motivate more in a bit, our primary aim is to estimate latent (input) variables from observed input variables measured with error and output variables connected to the inputs via GPs; a challenge leading to what are called latent GPs [49]. When using derivative GPs, such latent inputs are shared between the original outputs and their derivative counterparts, effectively doubling the amount of information available for estimating the latent inputs.

In real-world data, the derivatives are seldom exactly computed, which adds a major challenge to the modeling endeavor. If derivatives are computed with respect to the observed (non-latent) inputs, they are naturally only an approximation of the derivatives with respect to latent inputs. Conversely, if the derivatives are (implicitly) computed with respect to the latent inputs, the uncertainty of the latter will induce hard-to-quantify uncertainty in the estimated derivatives. One way or another, this lack of exact derivative information poses a serious challenge for derivative GP modeling. Existing approaches are not equipped to deal with the significant scale differences between outputs and derivatives, thus requiring modifications in its covariance functions to ensure valid and efficient latent variable estimation.

In this paper, we demonstrate a combination of all the above model extensions leading to our DGP-LVM: derivative Gaussian process latent variable model framework. We provide more context about the real-world modeling challenges in Section 3.1.1 as a basis for our motivation to develop DGP-LVM. The remainder of the paper is structured as follows. We discuss and provide context on related works in Section 3.2. We introduce our methodology and model development in Section 3.3 and perform extensive simulation studies in Section 3.4 that demonstrate the relevance of our contributions. We further illustrate DGP-LVM on a real single-cell RNA sequencing data in Section 3.5 before discussing our methods' limitations and future work in Section 3.6.

3.1.1 Motivation

In developmental biology, to describe temporal biological processes, researchers use stochastic approaches to understand cellular progression, that is, how cells develop and undergo changes in their state throughout various stages over the period of time [54, 55, 68]. Currently, this problem is frequently tackled with single-cell RNA sequencing (scRNA-seq) technologies by analyzing messenger-RNA (mRNA) molecule counts as a measure of gene expression [39]. The measured gene expression (also called expression levels) provide the necessary information about the nature of cells at a specific point of time, also known as cell states, as well as their changes over time. However, due to the experimental limitations of the current sequencing methods each cell gets destroyed in the measurement process and can therefore be observed only once. This situation makes it difficult to infer cell state transitions and the overall sequence of cell states of a temporal biological process. To that end, pseudotime ordering is a popular approach to describe such a biological process as a sequence of cell states along a time sequence [90].

Single-cell gene expression data provides information about cell state snapshots. While conventional pseudotime ordering approaches operate only on cell state snapshots to estimate pseudotime, only recently, directional information about cell state changes (i.e., derivative information) has been accounted for in this task [37]. Here, we hypothesize and demonstrate empirically later that including directional information on cell state transitions increases the precision in estimating pseudotime. This directional information is available through a quantity known as RNA velocity that is estimated from the difference in unspliced and spliced gene expression levels over latent experimental time (not to be confused with pseudotime) [9, 47]. By construction, this RNA velocity estimates the derivative of spliced gene expression data with respect to time. Concretely, our aim is to enable using the combination of RNA gene expression and RNA velocity in a single probabilistic framework for pseudotime estimation. This combination of RNA gene expression and its corresponding RNA velocity requires a novel statistical model approach.

In order to model such data, certain requirements must be satisfied. Starting from support for multi-dimensional outputs that allows inclusion of several genes for each cell, the model should account for varying gene-specific information as well as possible biologically induced interactions among genes. Moreover, since RNA velocity is only a derivative *estimation* of gene expression levels, they are frequently on a significantly different scale than the gene expression levels. Dealing with this scale difference is a challenge that the model must address in order to provide reliable pseudotime estimates. In this paper, we demonstrate that DGP-LVM is able to tackle all of the above challenges and can estimate latent input variables with significantly higher accuracy than other GP models. Thus, we also demonstrate its potential to be applicable to estimating pseudotime through RNA gene expressions and their corresponding RNA velocities.

3.1.2 Contributions

- We develop a probabilistic GP modeling framework for latent (input) variable estimation using derivative information for any multi-dimensional data-generating process. Our model accounts for dimension specific information and interactions between dimensions

in a multi-dimensional data scenario which are common in (but not limited to) fields like single-cell biology.

- We develop a custom derivative structure for Squared Exponential (SE) and Matérn class of covariance functions that is able to account for significant scale differences between the outputs and its corresponding derivatives.

- Through extensive simulations, we demonstrate that our model provides substantially more accurate latent variable estimates than other GP models under realistic scenarios.

- We showcase the application of our modeling approach on a reduced real-world scRNA-seq data set.

3.2 Related work

Gaussian processes, as a class of models, underwent a wide range of extensions over the years giving rise to various forms of GP models. Specifically, three broad extensions relevant to this work are GPs with derivative information, multi-output GPs and GPs for latent variable modeling. Using derivative information for Gaussian processes was introduced in [77] who replaced standard covariance functions with their derivative counterparts. This paved the way to modeling data along with its derivatives as a single GP model. Recently, derivative GPs were extended to support multiple inputs and scalable approximations [27, 65]. In case of multi-output GPs, recent works [43, 59] study GPs with support for multiple outputs that are of varying nature in terms of data types, however multi-output GPs with derivative information have not been studied in detail yet. Latent GPs were introduced by [49, 50] and have, so far, been predominantly used for dimensionality reduction [87, 88]. More recently, for similar applications of dimension-reduction technique, extensions on GP latent variable model for non-Gaussian likelihoods with different types of latent input structures were discussed in [48]. These works also focus on scalable approximations to latent GPs. In contrast, we focus on estimating latent inputs that probabilistically explains a dependent multi-output variable.

For the modeling of scRNA-seq data, GPs have been broadly applied in two relevant directions, specifically, for clustering [15, 16] and temporal modeling [41]. Considering the latter, pseudotime estimation constitutes a major research direction as it is directly related to understanding the true underlying biological processes. It has been shown previously that point estimates of pseudotime are highly prone to infer false cellular ordering, thus suggesting Bayesian inference to provide uncertainty estimates alongside each estimated pseudotime [20]. Further works focus on latent pseudotime estimations [21, 70] along with branching structures for trajectory inference [3] based on a GP framework. One of the main limitations in these works lie in their restricted use of gene expressions, taking into account expression profile snapshots as the only available information regarding cellular ordering. We provide evidence that including RNA velocity as derivative information holds the power to estimate latent pseudotime with increased precision compared to what previous approaches could achieve.

3.3 Methods

We develop DGP-LVM, a framework for derivative Gaussian process modeling with the primary goal of estimating latent variables serving as (implicit) inputs to the GP. As the general setup, we consider a pair of variables $(\mathbf{y}, \mathbf{x})$ where $\mathbf{y}$ is the output (response) variable and $\mathbf{x}$ is the input variable (covariate), with individual observations denoted as $y_i, x_i \in \mathbb{R}, i = \{1, ..., N\}$ where N is the number of observations. In addition to $\mathbf{y}$ itself, we incorporate the derivative outputs $\mathbf{y}^{(1)} = \delta\mathbf{y}/\delta\mathbf{x}$ into the model. The components of DGP-LVM are first discussed individually, before we combine them into a single model.

3.3.1 Derivative Gaussian processes

A GP is a stochastic process specified by a mean function $m = m(x)$, and a covariance function $k = k(x, x')$ with $x, x' \in \mathbb{R}$ such that a finite set of these points will follow a multivariate Gaussian distribution [95]. Concretely, we consider GPs $f(x)$ such that $f(x) \sim \mathcal{GP}(m, k)$. Here we consider a constant mean function (similar to an intercept in regression models). If the output variable $\mathbf{y}$ is univariate, modeling the relationship of $\mathbf{x}$ and $\mathbf{y}$ via a (single-output) GP and independent additive noise can be written as

$$y_i = f(x_i) + \varepsilon_i, \tag{3.1}$$

where ε_i is the i^{th} sample of $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ assuming equal-variance Gaussian noise. Together, this is equivalent to

$$y_i \sim \mathcal{N}(f(x_i), \sigma^2). \tag{3.2}$$

For, $i \neq j$ we have $\text{Cov}(y_i, y_j) = K(x_i, x_j)$ and for $i = j$, we have $\text{Cov}(y_i, y_j) = \text{Var}(y_i) = K(x_i, x_j) + \sigma^2$. The above notation will be extended to multi-output GPs in Section 3.3.2.

GPs are able to take advantage of derivatives in addition to its corresponding sample data to increase model accuracy. Since differentiation is a linear operator, a derivative of a GP is just another GP [69, 77]. This property of GPs can be utilized to take derivative of a joint covariance structure of both $\mathbf{y}$ and $\mathbf{y}^{(1)}$, if the second order derivative of the covariance function exists. The (joint) derivative GP is then given by

$$\begin{pmatrix} f(x) \\ f^{(1)}(x) \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} m_f \\ m_{f^{(1)}} \end{pmatrix}, \begin{pmatrix} k & k^{(1,0)} \\ k^{(0,1)} & k^{(1,1)} \end{pmatrix} \right), \tag{3.3}$$

where m_f and $m_{f^{(1)}}$ are constant mean functions corresponding to GP $\mathbf{f}$ and it's derivative $\mathbf{f}^{(1)}$ respectively. $k^{(1,0)}$ is the first derivative of the covariance function $k = K(x, x')$ with respect to x and $k^{(0,1)}$ is the first derivative of the covariance function with respect to x'. $k^{(1,1)}$ is the second order partial derivative of k differentiating both with respect to x and x'. In other words, differentiation simply propagates through the covariance function (see Appendix A.1 for mathematical details). The specific properties of such derivative GP models depend on the chosen covariance function. A common choice is the Squared Exponential (SE) covariance function with hyperparameters ρ as length scale and α as the GP marginal standard deviation

(SD). The derivative version of the SE covariance function is given by

$$k(x_i, x_j) = \alpha^2 \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right), \tag{3.4}$$

$$k^{(0,1)}(x_i, x_j) = \alpha^2 \frac{(x_i - x_j)}{\rho^2} \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right), \tag{3.5}$$

$$k^{(1,1)}(x_i, x_j) = \frac{\alpha^2}{\rho^4}(\rho^2 - (x_i - x_j)^2) \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right). \tag{3.6}$$

Derivative covariance functions are obtainable generally for any chosen covariance function whose second order derivative exists. In this paper, we focus on SE and Matérn class covariance functions as perhaps the most common choices. We provide further details on the mathematical forms of derivative SE, Matérn 3/2 and Matérn 5/2 covariance functions in Appendix A.1.

Customised hyperparameters

Properly including the derivative observations $\mathbf{y}^{(1)}$ requires more than just using a basic derivative covariance function. Due to the properties of differentiation, $\mathbf{y}^{(1)}$ can be on a fundamentally different scale than $\mathbf{y}$ and thus needs to be treated as such. In addition to having different signals (i.e., different GP components $f(x)$ vs. $f^{(1)}(x)$), the error SDs for $\mathbf{y}$ and $\mathbf{y}^{(1)}$ will also be different. Moreover, for real data, the observations $\mathbf{y}^{(1)}$ containing derivative information may not be the exact same as the true derivatives $\delta\mathbf{y}/\delta\mathbf{x}$, but only be proportional to them (see Section 3.1). This proportionality induces a scale difference between $\mathbf{y}^{(1)}$ and what is canonically modeled by a basic derivative covariance function. This creates a major issue for models ignoring scale differences as we demonstrate in our simulations.

To incorporate these scale considerations into our model, we propose to adjust the covariance function hyperparameters. Again using the SE covariance function as an example, we propose to introduce a second marginal SD parameter $\alpha^{(1)}$, corresponding to the derivative part of the GP, while α now only concerns the original part of the GP:

$$k(x_i, x_j) = \alpha^2 \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right), \tag{3.7}$$

$$k^{(0,1)}(x_i, x_j) = \alpha\alpha^{(1)} \frac{(x_i - x_j)}{\rho^2} \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right), \tag{3.8}$$

$$k^{(1,1)}(x_i, x_j) = \frac{{\alpha^{(1)}}^2}{\rho^4}(\rho^2 - (x_i - x_j)^2) \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right). \tag{3.9}$$

In other words, since $k(x_i, x_j) = Cov(y_i, y_j)$ we account for the GP marginal variance through α^2 since we are only concerned with the original part of the GP $\mathbf{f}$. In the case of $Cov(y_i, y_j^{(1)})$ and $Cov(y_i^{(1)}, y_j)$, we compute $k^{(0,1)}(x_i, x_j)$ and thus we account for the GP marginal variance through $\alpha\alpha^{(1)}$ where α belongs to $\mathbf{f}$ and $\alpha^{(1)}$ belongs to $\mathbf{f}^{(1)}$. Finally, in the case of $Cov(y_i^{(1)}, y_j^{(1)})$, we compute $k^{(1,1)}(x_i, x_j)$ and are thus accounting for the GP marginal variance through ${\alpha^{(1)}}^2$ since we are only dealing with the derivative part of the GP $\mathbf{f}^{(1)}$. Since, in addition to the

product term $\alpha\alpha^{(1)}$ in $k^{(0,1)}$, we also estimate α^2 and ${\alpha^{(1)}}^2$ separately through k and $k^{(1,1)}$, respectively, estimating both α and $\alpha^{(1)}$ does not cause identifiability issues.

Similarly, we define two residual standard deviation parameters σ and $\sigma^{(1)}$, accounting for measurement noise in $\mathbf{y}$ and $\mathbf{y}^{(1)}$, respectively. The scale of ρ is only dependent on the scale of $\mathbf{x}$, which is constant across outputs and their derivatives, such that ρ does not need to be split up into two parameters. Together, independent of specifically chosen covariance function, the DGP-LVM on $\mathbf{y}$ and $\mathbf{y}^{(1)}$ with independent, additive Gaussian noise is then specified as

$$y_i \sim \mathcal{N}(f(x_i), \sigma^2) \quad \text{and} \quad y_i^{(1)} \sim \mathcal{N}(f^{(1)}(x_i), {\sigma^{(1)}}^2). \tag{3.10}$$

3.3.2 Multidimensional outputs

Multivariate output GPs (or multi-output GPs) model multiple response variables $\{\mathbf{y}_1, ..., \mathbf{y}_D\}$ jointly over $D > 1$ output dimensions [69]. Extending our univariate notation, the individual output values are now denoted as y_{di} for dimension d and observation i, with corresponding derivative values $y_{di}^{(1)}$. Multi-output GPs are created by first setting up D independent, univariate Gaussian processes $f_d(x)$ each with their own set of hyperparameters, that is, (ρ_d, α_d, $\alpha_d^{(1)}$, σ_d and $\sigma_d^{(1)}$) for Matérn class of covariance functions with adjusted scales. Subsequently the univariate GPs are related to one another by folding them with a (D-dimensional) across-dimension correlation matrix $\mathbf{C}$ [12, 85]. That is, for each observation i, we obtain a vector of across-dimension correlated GP values as

$$\begin{pmatrix} \tilde{f}_1(x_i) \\ \dots \\ \tilde{f}_D(x_i) \end{pmatrix} = \mathbf{A} \times \begin{pmatrix} f_1(x_i) \\ \dots \\ f_D(x_i) \end{pmatrix}, \tag{3.11}$$

where $\mathbf{A}$ is the Cholesky factor of $\mathbf{C}$, that is, $\mathbf{C} = \mathbf{A}\mathbf{A}^T$ with $\mathbf{A}$ being lower-triangular. This way, multi-output GPs combine two dependency structures, one within dimensions (and across observations) as expressed by the univariate GPs through corresponding covariance functions and one across output dimensions (but within observations) as expressed by $\mathbf{C}$ (or $\mathbf{A}$).

This readily generalizes to derivative GPs by applying Equation (3.11) to the derivative GP values $f_d^{(1)}(x_i)$ as well, which results in the across-dimension correlated values $\tilde{f^{(1)}}_d(x_i)$. Adding independent Gaussian noise, our derivative multi-output GP model then implies for all d and i:

$$y_{di} \sim \mathcal{N}(\tilde{f}_d(x_i), \sigma_d^2) \quad \text{and} \quad y_{di}^{(1)} \sim \mathcal{N}(\tilde{f^{(1)}}_d(x_i), {\sigma_d^{(1)}}^2). \tag{3.12}$$

3.3.3 Latent variable inputs

So far, we have considered the input $\mathbf{x}$ to be known exactly. However, in practice, we often only have a noisy measurement $\tilde{\mathbf{x}}$ of $\mathbf{x}$ available. In this context, the true $\mathbf{x}$ becomes a latent variable, which needs to be appropriately modeled and subsequently estimated. If we assume that the measurements $\tilde{\mathbf{x}}$ are Gaussian with known measurement SD s, we can write for each

observation i:

$$\tilde{x}_i \sim \mathcal{N}(x_i, s^2). \tag{3.13}$$

The implied latent x_i is then passed to the GP covariance function, which results in what is known as latent(-input) GPs [49, 50, 88]. Such latent-input GPs are even harder to fit than their non-latent counterparts: Not only does the number of unknowns increase substantially, but also new identification issues arise due to both $\mathbf{x}$ and $\boldsymbol{\rho} = \{\rho_1, \dots, \rho_D\}$ now being unknown (see Section 3.3.4 for details on how we deal with this).

3.3.4 The full model

Below, we summarize all the extensions that together make up our proposed DGP-LVM framework. To shorten the notation, let us denote the vector of GP hyperparameters for dimension d as $\boldsymbol{\theta}_d$. For our considered covariance functions, $\boldsymbol{\theta}_d$ includes the length scale ρ_d, GP marginal SDs α_d and $\alpha_d^{(1)}$ as well as the error SDs σ_d and $\sigma_d^{(1)}$. Considering the SE covariance function as an example, full DGP-LVMs are then specified as follows:

$$\begin{aligned}
\begin{pmatrix} f_d(x) \\ f_d^{(1)}(x) \end{pmatrix} &\sim \mathcal{GP}\left(\begin{pmatrix} m_{f_d} \\ m_{f_d^{(1)}} \end{pmatrix}, \begin{pmatrix} k_d & k_d^{(1,0)} \\ k_d^{(0,1)} & k_d^{(1,1)} \end{pmatrix} \right), \\
k_d(x_i, x_j) &= \alpha_d^2 \exp\left(-\frac{(x_i - x_j)^2}{2\rho_d^2} \right), \\
k_d^{(0,1)}(x_i, x_j) &= \alpha_d \alpha_d^{(1)} \frac{(x_i - x_j)}{\rho_d^2} \exp\left(-\frac{(x_i - x_j)^2}{2\rho_d^2} \right), \\
k_d^{(1,1)}(x_i, x_j) &= \frac{{\alpha_d^{(1)}}^2}{\rho_d^4} (\rho_d^2 - (x_i - x_j)^2) \exp\left(-\frac{(x_i - x_j)^2}{2\rho_d^2} \right), \\
y_{di} &\sim \mathcal{N}(f_d(x_i), \sigma_d^2), \\
y_{di}^{(1)} &\sim \mathcal{N}(f_d^{(1)}(x_i), {\sigma_d^{(1)}}^2), \\
\tilde{x}_i &\sim \mathcal{N}(x_i, s^2), \\
\boldsymbol{\theta}_d &\sim p(\boldsymbol{\theta}_d) = p(\rho_d)\, p(\alpha_d)\, p(\alpha_d^{(1)})\, p(\sigma_d)\, p(\sigma_d^{(1)}).
\end{aligned} \tag{3.14}$$

Following the above specifications, after marginalizing out $\mathbf{f}$ and $\mathbf{f}^{(1)}$, the multi-output joint probability density factorizes as

$$p(\mathbf{y}, \mathbf{y}^{(1)}, \mathbf{x}, \boldsymbol{\theta}) = \prod_d^D p(\mathbf{y}_d \mid \mathbf{x}, \boldsymbol{\theta}_d)\, p(\mathbf{y}_d^{(1)} \mid \mathbf{x}, \boldsymbol{\theta}_d)\, p(\mathbf{x})\, p(\boldsymbol{\theta}_d). \tag{3.15}$$

where $p(\mathbf{y}_d \mid \mathbf{x}, \boldsymbol{\theta}_d)$ and $p(\mathbf{y}_d^{(1)} \mid \mathbf{x}, \boldsymbol{\theta}_d)$ denote the respective GP-based likelihoods for a single output dimension. $p(\mathbf{x})$ denotes the prior for the latent $\mathbf{x}$ implied by the measurement model Eq.(3.13). More details on the choice of prior distributions are discussed in Section 3.4.2. Using

Bayes' rule, we obtain the joint posterior over $\mathbf{x}$ and $\boldsymbol{\theta}$ as

$$p(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}, \mathbf{y}^{(1)}) = \frac{p(\mathbf{y}, \mathbf{y}^{(1)}, \mathbf{x}, \boldsymbol{\theta})}{\int \int p(\mathbf{y}, \mathbf{y}^{(1)}, \mathbf{x}, \boldsymbol{\theta})\, d\mathbf{x}\, d\boldsymbol{\theta}}. \tag{3.16}$$

Posterior samples of $\mathbf{x}$ and $\boldsymbol{\theta}$ (i.e. all the covariance function hyperparameters) are obtained through MCMC sampling via adaptive Hamiltonian Monte Carlo [42, 62]. We implemented all models in Stan using the RStan interface [80].

Implemented as above, DGP-LVMs can be applied to the aforementioned problem of pseudotime estimation from single-cell RNA sequencing data. The scRNA-seq data we consider consists of spliced RNA gene counts and RNA velocity, the time derivative of gene counts. DGP-LVM allows inclusion of both these information into a single model (see Section 3.3.1). Since the RNA velocity is not an exact derivative of spliced RNA counts, it induces a scale difference that is solved by DGP-LVM as shown in Section 3.3.1. Given that single-cell RNA sequencing data is multi-dimensional, DGP-LVM is designed as a multi-output model (see Section 3.3.2). The primary aim of DGP-LVM is to estimate the latent inputs as explained in Section 3.3.3, which perfectly aligns with pseudotime estimation since pseudotime is an unobserved cell ordering, and is hence considered as a latent variable. Moreover, the RNA sequencing data comes with its own cell capture time or experimental time which can be considered as a noisy version of the true pseudotime.

3.4 Simulation study

The fundamental issue for validating and comparing models designed to estimate latent variables is the lack of ground truth values for real-world data. Thus, it is crucial to test any latent variable model through extensive simulations where the ground truth is available and controllable. Below, we discuss and provide evidence for the importance of our proposed model innovations. Concretely, we showcase DGP-LVM on multiple simulated data setups that closely represent the complexities of a real scRNA sequencing data.

3.4.1 Simulated data

We consider five primary scenarios to generate simulated data. In our first scenario, we generate data from a multi-output GP with scaled derivative SE covariance function. In our second and third scenarios, we generate data from a multi-output GP but with scaled derivative Matérn 3/2 and 5/2 covariance functions, respectively. All of the above scenarios constitute cases where the estimated DGP-LVMs align with the true underlying process. However, datasets generated this way can vary strongly in the amount of signal they contain, thus adding a lot of random variation in the simulation results. To account for this issue, in our fourth scenario, we generate

data from a derivative periodic process with the true generating function

$$\begin{aligned} f_{ij} &= \alpha_j^2 \sin\left(\frac{x_i}{\rho_j}\right) \\ f_{ij}^{(1)} &= \frac{{\alpha_j^{(1)}}^2}{\rho_j} \cos\left(\frac{x_i}{\rho_j}\right) \end{aligned} \tag{3.17}$$

and corresponding data simulations as

$$y_{ij} \sim \mathcal{N}(f_{ij}, \sigma_j^2) \quad \text{and} \quad y_{ij}^{(1)} \sim \mathcal{N}(f_{ij}^{(1)}, {\sigma^{(1)}}_j^2). \tag{3.18}$$

In all of these scenarios, the data is generated with varying hyperparameters and correlated outputs assumptions in play (see Section 3.3). The hyperparameters of the periodic data generating process fulfill a similar purpose to those of SE, Matérn 3/2 and Matérn 5/2 derivative GPs. Hence we choose to use the same hyperparameter names for simplicity. The scenario of periodic data (Eq.(3.17)) is important on two counts. First, it allows us to better control the amount of signal contained in each generated dataset. Second, it demonstrates that DGP-LVM can achieve good results even when the underlying generating process in not actually a GP.

Additionally, in the fifth scenario, we further increase the complexity by adding a quadratic and linear trend to the above periodic and corresponding derivative functions, respectively, as

$$\begin{aligned} f_{ij} &= \alpha_j^2 \sin\left(\frac{x_i}{\rho_j}\right) + bx_i^2, \\ f_{ij}^{(1)} &= \frac{{\alpha_j^{(1)}}^2}{\rho_j} \cos\left(\frac{x_i}{\rho_j}\right) + 2bx_i, \end{aligned} \tag{3.19}$$

where b is a scaler and is chosen in accordance with the acting periodicity parameter $\boldsymbol{\rho}$. The data generating process then follows Eq.(3.18). This fifth scenario is designed to test the limitations of modeling non-stationary data with stationary GPs.

To demonstrate the adversity of scale difference between $\mathbf{y}$ and $\mathbf{y}^{(1)}$, we induce a scaling factor of $\lambda = 3$ that propagates through the GP marginal SD and error SD (see Section 3.3.1). The GP marginal SD for the output $\mathbf{y}$ and the derivative $\mathbf{y}^{(1)}$ are related through λ such that $\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha}^{(1)}$. Similarly for the error SD, $\boldsymbol{\sigma} = \lambda\boldsymbol{\sigma}^{(1)}$. Therefore, we only specify sampling distributions for $\boldsymbol{\alpha}^{(1)} \sim \text{Normal}^+(3, 0.25^2)$ and $\boldsymbol{\sigma}^{(1)} \sim \text{Normal}^+(1, 0.25^2)$. In reality, $\lambda > 3$ or $\lambda < 1/3$ may very well occur (see Section 3.5). In the simulations, we chose to avoid more extreme λ values to prevent substantial convergence issues for models without scaling modifications. This allows us to showcase these models' (reduced) performance without confounding this finding with convergence considerations. The ground truth GP length scale is sampled as $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.05^2)$. The choice of our sampling distributions, especially for the length scale $\boldsymbol{\rho}$ being an informed prior, enables us to explicitly select a range of values for our hyperparameters for which the simulated GP data would contain sufficient amount of signal. Further, we introduce an uniform between-dimension correlation of 0.5 for all the simulated data scenario as to represent

moderate interactions between outputs. Combined with the sampling of true hyperparameters for each output dimension, our simulated data mimics the real-world data scenario where such assumptions are prevalent.

Lastly, we generate ground truth for x as a sequence of values between $\{0.5, ..., 9.5\}$ with a total output sample size of $N = 20$ for $\mathbf{y}$ and $\mathbf{y}^{(1)}$ each and we choose a value for prior measurement SD of the noisy $\tilde{\mathbf{x}}$ as $s = 0.3$ (see Section 3.3.3). This resembles the realistic scenario where observed times are already a relatively good measure of latent pseudotime considering the overall input scale. Both simulation studies are generated as multi-output data with three sets of output dimensions namely $D = 2, 5$ and 10. To keep model estimation times within manageable bounds, each simulated dataset contains only 20 $\mathbf{y}$ and correspondingly 20 $\mathbf{y}^{(1)}$ sample points. We perform 50 trials for each simulation scenario, that is, generate 50 datasets for the three GP data, the periodic and the periodic with trend scenario, respectively.

3.4.2 Model setup

The models within our DGP-LVM framework can vary specifically in four components, namely the inclusion of (1) derivative information, (2) scaled derivatives, (3) varying hyperparameters, as well as (4) correlations across output dimensions. In order to study the individual importance of the four components, we systematically enable/disable each of them and investigate the resulting models' performances. The underlying data generating process contains all of the above components, so any model that only has a subset of components will be misspecified at least to some degree. In our simulations, component combinations were fully crossed where sensible (see Table 3.1 for an overview). We fit 12 GP models for each selected number of output dimensions D resulting in a total of 36 models fitted per generated dataset for a specific simulation scenario. We only exclude specific, non-sensible combinations. For example, it does not make sense to ask if a GP model, which does not include derivative information, accounts for the scaling of the derivatives. We use a constant mean function (similar to an intercept in regression models) in all our models for both $\mathbf{f}$ and $\mathbf{f}^{(1)}$(wherever applicable) throughout the simulation study. Such specifications help with overall location shifts in the data.

Prior specifications for all the model hyperparameters involved were aligned with the data generating conditions to a reasonable extent to better showcase the model contributions. We specify separate priors for the marginal SDs and error SDs corresponding to $\mathbf{f}$, $\mathbf{f}^{(1)}$ and $\mathbf{y}$, $\mathbf{y}^{(1)}$ respectively, to account for the scale differences between the original and derivative part of the data. We specify the priors for GP marginal SDs $\boldsymbol{\alpha} \sim \text{Normal}^{+}(9, 0.75^2)$ and $\boldsymbol{\alpha}^{(1)} \sim \text{Normal}^{+}(3, 0.25^2)$. In case of error SDs we specify $\boldsymbol{\sigma} \sim \text{Normal}^{+}(3, 0.75^2)$ and $\boldsymbol{\sigma}^{(1)} \sim \text{Normal}^{+}(1, 0.25^2)$. We shifted our priors for the original part of the GP $\mathbf{f}$ and data $\mathbf{y}$ in accordance with our choice of scale difference λ in the simulation scenarios. We use an informative prior on the length scale $\boldsymbol{\rho} \sim \text{Normal}^{+}(1, 0.05^2)$ is roughly based on the mean Euclidean distance between the $\tilde{\mathbf{x}}_i$ (as in Section 3.3.3) as well as the data generating specifications showed in Section 3.4.1. As prior for the between-output correlation matrix $\mathbf{C}$, we apply an unimodal LKJ(1) distribution [52] defined over the positive definite symmetric matrices with unit diagonals. This distribution is a common prior choice for correlation matrices. For our constant mean function (intercepts), we

Table 3.1: GP models along with their specifications used for simulated scenarios

Derivative information	Scaled derivatives	Varying hyperparameters	Correlated outputs
✓	✓	✓	✓
✓	✓	✓	✗
✓	✓	✗	✓
✓	✗	✓	✓
✓	✓	✗	✗
✓	✗	✗	✓
✓	✗	✓	✗
✓	✗	✗	✗
✗	✗	✓	✓
✗	✗	✓	✗
✗	✗	✗	✓
✗	✗	✗	✗

Note: Each table row denotes the assigned modifications to the fitted models. The first row shows the modifications involved in DGP-LVM.

used a Normal distribution with data-specific Mean and SD (for both $\mathbf{y}$ and $\mathbf{y}^{(1)}$ correspondingly) as prior.

All models were specified in Stan [80] and fitted with a single MCMC chain of 3000 iterations in total with 1000 warm-ups. We decided to run only a single chain per model to reduce overall computation times. However, we show that using multiple chains yield similar results down the line (see Appendix A.2: Figures A19 and A20.). All models were fitted on all generated datasets. The full study design is depicted in Figure 3.1.

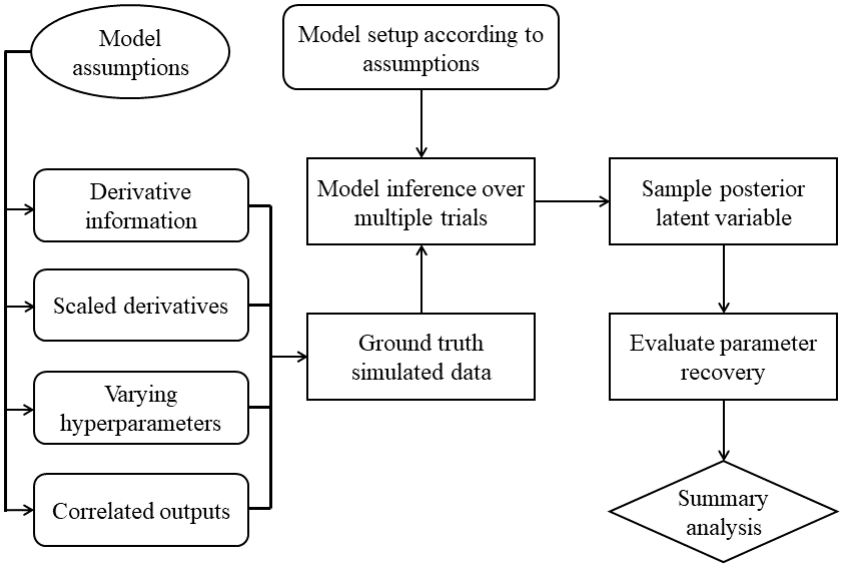

Figure 3.1: *High-level overview of the simulation study design.*

The simulation studies were conducted using 50 vCPUs (Intel(R) Xeon(R) Gold 6230R CPU @ 2.10 GHz) with 720 GB memory allowances. The maximum runtime (in hours) for DGP-LVM with $D = 10$ (for a simulated dataset) were approximately 4.6 for the SE model, 5.9 for the

Matérn 3/2 model and 8.9 for the Matérn 5/2 model.

3.4.3 Summary methods

In order to evaluate how well fitted models recover the latent ground truth, we compare posterior samples of the latent input variable $\mathbf{x}$ denoted by $\mathbf{x}_{post}$ with their respective ground truth values denoted by $\mathbf{x}_{true}$ using the root mean squared error (RMSE):

$$\text{RMSE}(\mathbf{x}_{post}) = \sqrt{\mathbb{E}(\mathbf{x}_{post} - \mathbf{x}_{true})^2} \tag{3.20}$$

and the mean absolute error:

$$\text{MAE}(\mathbf{x}_{post}) = \mathbb{E}(\mid \mathbf{x}_{post} - \mathbf{x}_{true} \mid) \tag{3.21}$$

where the expectations are taken over the posterior (approximated via samples). In case of RMSE, $\mathbb{E}(\mathbf{x}_{post} - \mathbf{x}_{true})^2$ can be decomposed into $\text{Var}(\mathbf{x}_{post})$ and $\text{Bias}(\mathbf{x}_{post}, \mathbf{x}_{true})^2$, thus measuring Bias-variance trade-off. We compute RMSE and MAE from all fitted models shown in Table 3.1 for each set of output dimensions (2,5 and 10). We prefer models that provide both low bias indicating posterior mean estimates close to the ground truth as well as lower posterior variance indicating high precision, together resulting in an overall low RMSE. Similarly, we prefer low MAE since it shows the amount of absolute bias present while estimating the latent variable. Overall, RMSE penalizes the models for estimating outlying posteriors while MAE is more lenient in that sense.

To analyze RMSE and MAE values, we use a multilevel analysis of variance model (ANOVA) fitted with brms [17], which disentangles the contributions of each model component. Using a multilevel model is important to account for the dependency between results of all models fitted on the same dataset. We model fixed main effects of scaled derivatives, varying hyperparameters, correlated outputs and number of output dimensions. For this purpose, we consider scaled derivatives as a factor variable with three levels corresponding to models that (a) do not include derivative information, (b) models that include derivative information with scaling and (c) models that include derivative information without scaling. Varying hyperparameters are represented by a binary factor variable that denotes varying vs. constant hyperparameters across output dimensions. Similarly, correlated outputs represented by a binary factor variable that indicates if the multiple outputs are assumed to be correlated or not. We additionally model fixed interaction effects between (a) scaled derivatives and varying hyperparameters and (b) scaled derivatives and correlated outputs. Since our simulation study is performed over 2, 5, and 10 output dimensions, we include dimension as factor variable with three levels and allowed it to interact with all previously mentioned (fixed) main and interaction effects. We account for the dependency structure in the RMSE and MAE values, induced by fitting multiple models to the same simulated dataset, by a random intercept over datasets as well as corresponding random slopes of the scaled derivatives, varying hyperparameters, and correlated outputs factors. Further, we account for the dependency in the evaluation metric values for the 20 latent inputs estimated from a single model through a random intercept per fitted model. The results based

on RMSE are presented in Section 3.4.5 while their corresponding MAE results are shown in Appendix A.2.

3.4.4 Model convergence

We investigate the convergence of our fitted GP models for all the five simulation scenarios mentioned before. To that end, we use standard MCMC sampling diagnostics including state-of-the-art versions of the scale reduction factor $\widehat{R}$, the bulk effective sample size (Bulk-ESS) and the tail effective sample size (Tail-ESS) [91]. The combined check of these measures provide a comprehensive picture of individual parameter model convergence.

In general, $\widehat{R}$ should be very close to 1 and should ideally not exceed 1.01 [91]. In a simulation setup, we can evaluate the goodness of the posterior estimation also independently of convergence, as we have access to ground truth values. Hence, also in light of the relatively short MCMC chains, we decide to apply a more relaxed threshold of 1.1. Bulk-ESS indicates the reliability of measures of central tendency such as the posterior mean or median. Tail-ESS indicates the reliability of the 5% and 95% quantile estimates, which are commonly used to construct credible intervals. Both Bulk-ESS and Tail-ESS should have values greater than 100 times the number of MCMC chains. We computed all the convergence measures with the posterior package [19].

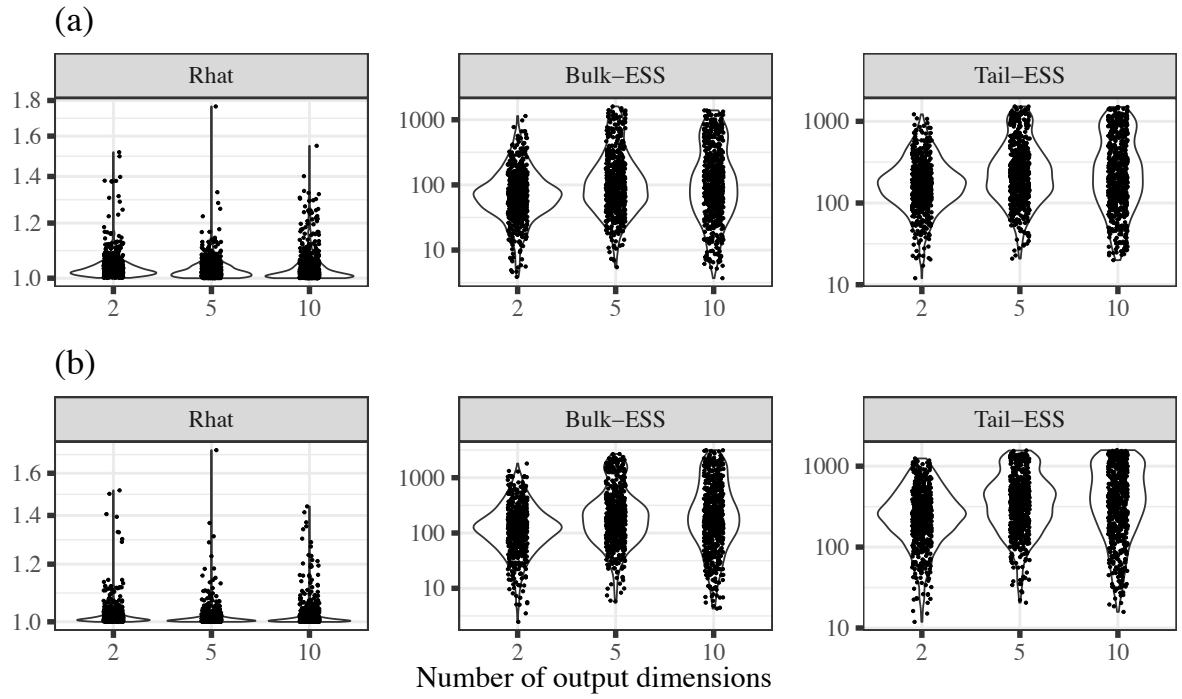

Figure 3.2: *Squared exponential scenario: Convergence measures for (a) latent inputs and (b) GP hyperparameters. The individual points correspond to each fitted models per simulated data. The y-axis for Bulk and Tail ESS plots are log10 transformed.*

For latent inputs and hyperparameters obtained from the simulated SE data scenario, we show $\widehat{R}$, Bulk-ESS, and Tail-ESS in Figure 3.2. We present the MCMC diagnostics plots for the other simulated data scenarios (see Figures A1-A4) in Appendix A.2 owing to their similar nature.

The $\widehat{R}$ were all satisfactory for majority of the simulation trials, with the exception of a few outlying models per trial. The GPs with derivative SE covariance function per simulation trial had better convergence as compared to the GPs with derivative Matérn covariance functions. This was expected due to the increased model complexity of the Matérn covariance functions. Moreover, the derivative Matérn 3/2 (see Figure A1) being the boundary of existing derivative covariance functions among the Matérn class makes it more complex for the sampler to perform as good as the Matérn 5/2 (see Figure A2) and subsequently the much simpler SE covariance function. The Bulk-ESS and Tail-ESS were consistently higher than the suggested threshold for all the cases, thus satisfying the recommended criteria. The convergence results for the periodic and periodic with trend scenarios as shown in Figures A3 and A4 were similar to derivative SE data simulation scenario.

3.4.5 Results

For all of the simulation scenarios discussed in Section 3.4.1, we evaluated the effects of including derivative information, accounting for scale differences between $\mathbf{y}$ and $\mathbf{y}^{(1)}$, estimating varying hyperparameters across multiple outputs as well as correlated outputs. We summarize these aforementioned conditions as model assumptions and show their effects on the RMSE as our primary model evaluation measure of the posterior estimates of latent $\mathbf{x}$ and covariance function hyperparameters with respect to their true simulated values. The corresponding MAE results were qualitatively highly similar and are thus only shown in Appendix A.2.

Model evaluation: Latent inputs

In addition to the posterior model evaluation measure estimates obtained from multilevel ANOVA, we also show the prior RMSE that would be expected if we only used the prior measurement model $\tilde{x}_i \sim \mathcal{N}(x_i, s^2)$ to infer $\mathbf{x}$. Consequently, the prior evaluation measure acts as a benchmark to illustrate how much precision we gain through the GP modeling of output data.

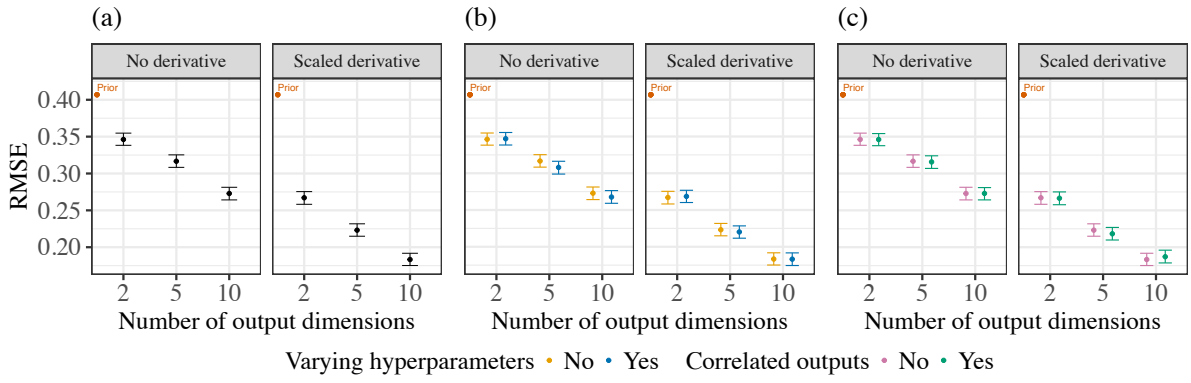

Figure 3.3: *Squared exponential scenario: Main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs*

Our findings for the latent $\mathbf{x}$ are presented for different simulation scenarios in Figures 3.3–3.5 for the simulated GP data scenarios and in Figure 3.6 for the periodic data scenario. We

see how the inclusion of both derivative information and scaling modifications simultaneously results in an overall substantial decrease in mean RMSE in the simulated SE and periodic data scenarios (Figure 3.3(a) and 3.6(a)), thus indicating a better recovery of the true latent values as compared to models without derivative information. In case of the Matérn 3/2 and 5/2 data scenarios, although not as substantial as the SE and periodic case, we see similar effects of adding scaled derivatives. This is due to the more challenging nature of the GPs with derivative Matérn covariance functions as seen through their model convergence. Additionally, the evaluation measures for all the scenarios further decrease as we increase the number of output dimensions.

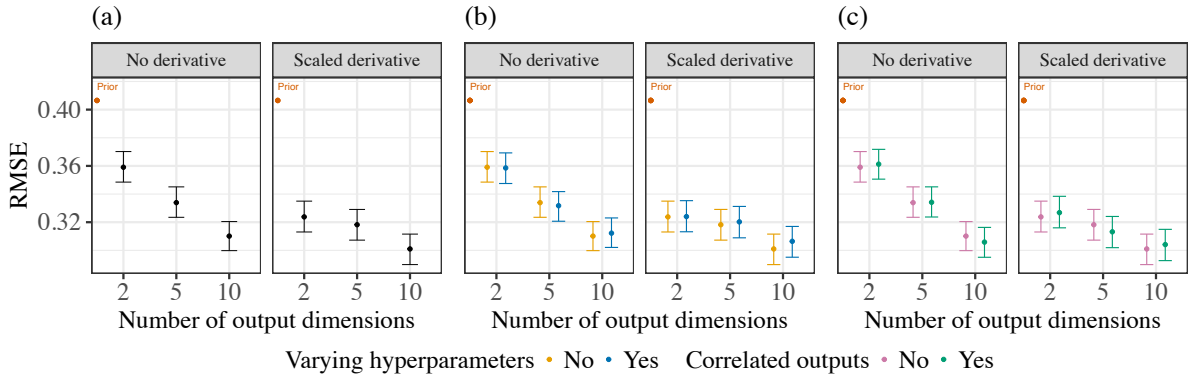

Figure 3.4: *Matérn 3/2 scenario: Main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs*

Overall, we see a reduction in RMSE of more than 50% compared to the corresponding prior metrics, and a reduction of about 30% compared to the models without derivative information in the SE and periodic data cases, thus clearly outlining the benefits of using DGP-LVMs. Conversely, when models include derivative information *without* accounting for scale differences, the RMSEs are a lot higher, suggesting that the model performs adversely while estimating latent inputs (see Figures A5–A8 in Appendix A.2). Curiously, the performance of such models is even worse than the models not including derivatives at all, sometimes close to (or even worse than) when just using the prior measurement model alone. Presumably, this is because hyperparameter estimates are strongly biased if forced to be the same for both regular outputs and their derivatives; at least when the ground truth assumes hyperparameters to be different by a factor of 3 (which is not unrealistic). As an implication, we then also obtain strongly biased latent input estimates, resulting in large RMSEs. This clearly highlights the importance of our derivative covariance function modifications. Without these modifications, using derivative information poses the risk of providing strongly misleading results.

With respect to the other varied components, modeling varying hyperparameters and correlated outputs may result in a slight increase in the RMSEs (see (b) and (c) of Figures 3.4–3.6), especially in higher output dimensions. We hypothesize that this is due to the significant increase in the number of estimated parameters, while the amount of data points remained constant in our simulations. Concretely, the number of parameters increase by the number of hyperparameters per dimension (i.e., 5 in our case) times the number of output dimensions D,

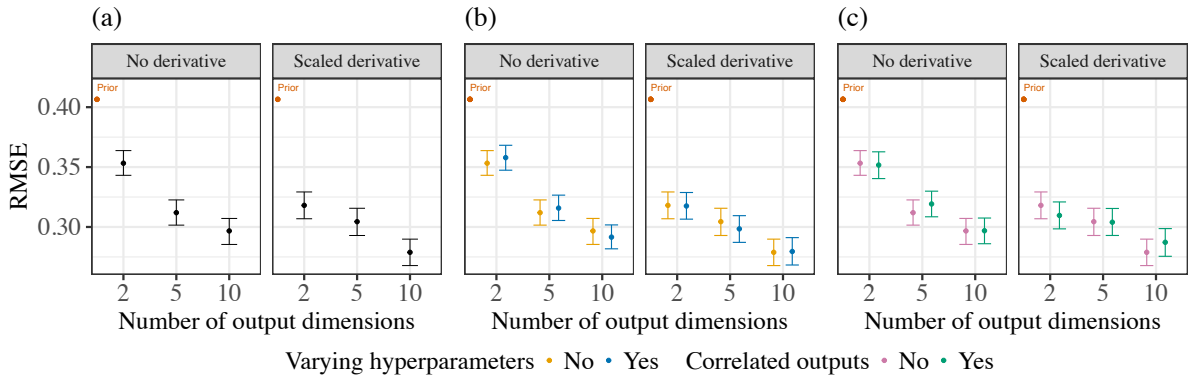

Figure 3.5: *Matérn 5/2 scenario: Main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs*

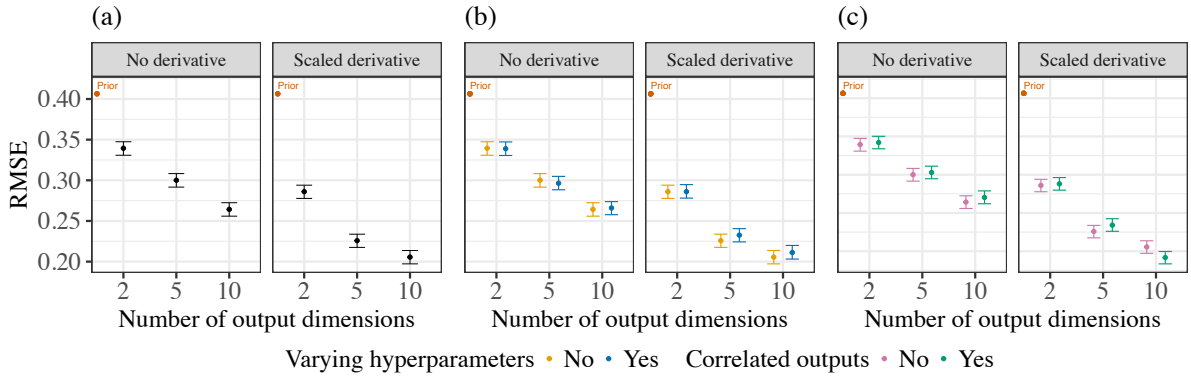

Figure 3.6: *Periodic scenario: Main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs*

which is quite substantial already for $D = 10$ output dimensions.

We encounter a similar issue when modeling outputs as correlated since the increase in estimated model parameters are even more substantial. For output dimension D, we estimate $D(D-1)/2$ number of parameters just for between-dimension correlations. Such a significant increase in parameters becomes visible in the results especially for $D = 10$ for most of the simulated scenarios.

Model evaluation: Hyperparameters

In Figures 3.7–3.10, we show hyperparameter recovery for the full DGP-LVM in the GP simulation scenarios and the periodic simulation scenario and compare the effects of accounting for scaling (each figure shows (a) full model with scaling and (b) without scaling).

For most of the simulated datasets, the hyperparameters show good recovery as indicated by low RMSE. We do see some extreme RMSE values though, especially for GP length-scales $\boldsymbol{\rho}$. These extreme cases are explained by the significant increase in the number of estimated parameters, when we consider both varying hyperparameters and correlated outputs without increasing the amount of data.

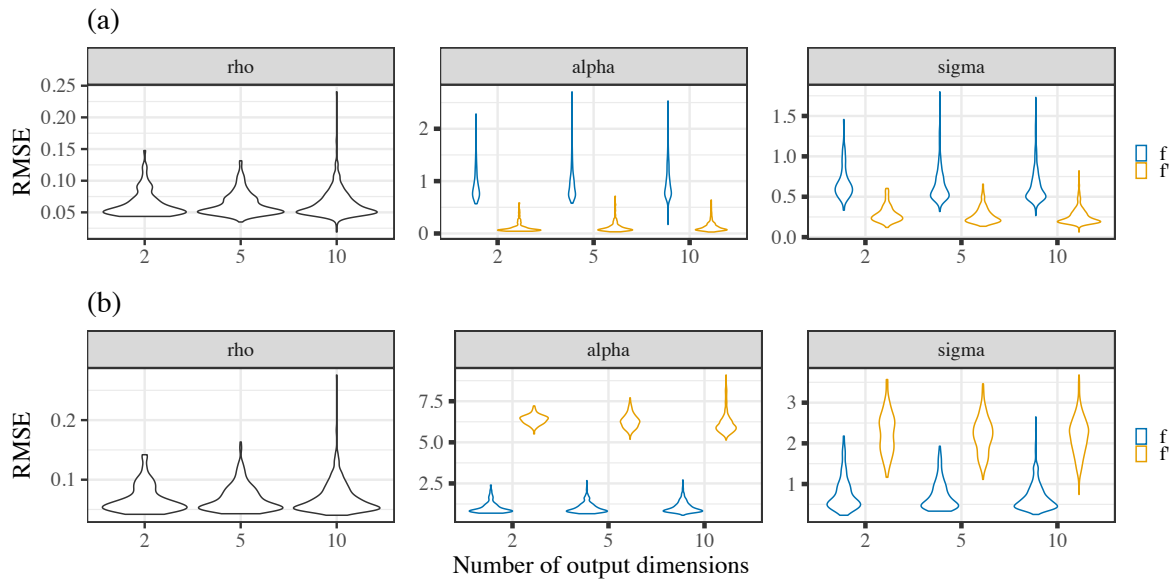

Figure 3.7: *Squared exponential scenario: Hyperparameter RMSEs for (a) full DGP-LVM model and (b) models without scale assumption. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

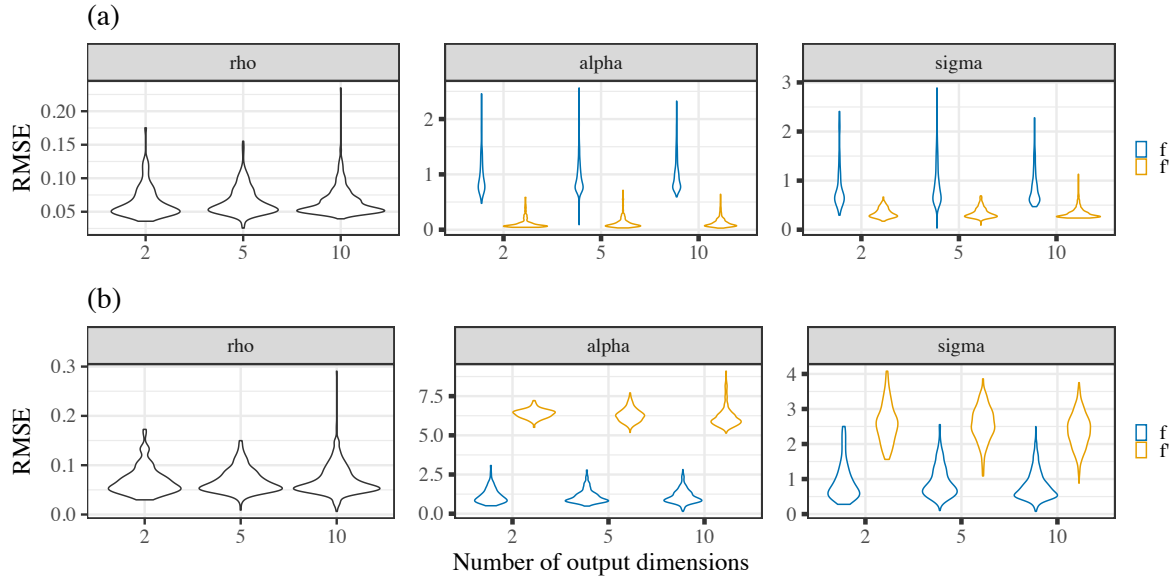

Figure 3.8: *Matérn 3/2 scenario: Hyperparameter RMSEs for (a) full DGP-LVM model and (b) models without scale assumption. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

Interestingly, we see how the scaling assumption helps identifying the GP marginal SD $\boldsymbol{\alpha}^{(1)}$ and error SD $\boldsymbol{\sigma}^{(1)}$ for the $\mathbf{f}^{(1)}$ and $\mathbf{y}^{(1)}$ respectively. Without the assumption, the model simply fails to recover the true SD hyperparameters for the derivative part of the data. Additionally, when disabling the assumptions of varying hyperparameters, correlated outputs separately as well as together (see (a), (b) and (c) of Figures A15–A18 respectively in Appendix A.2), we see how recovery of hyperparameters, especially the GP marginal SDs struggle due to model misspecification. This demonstrates that each of the model innovations discussed in Section 3.3

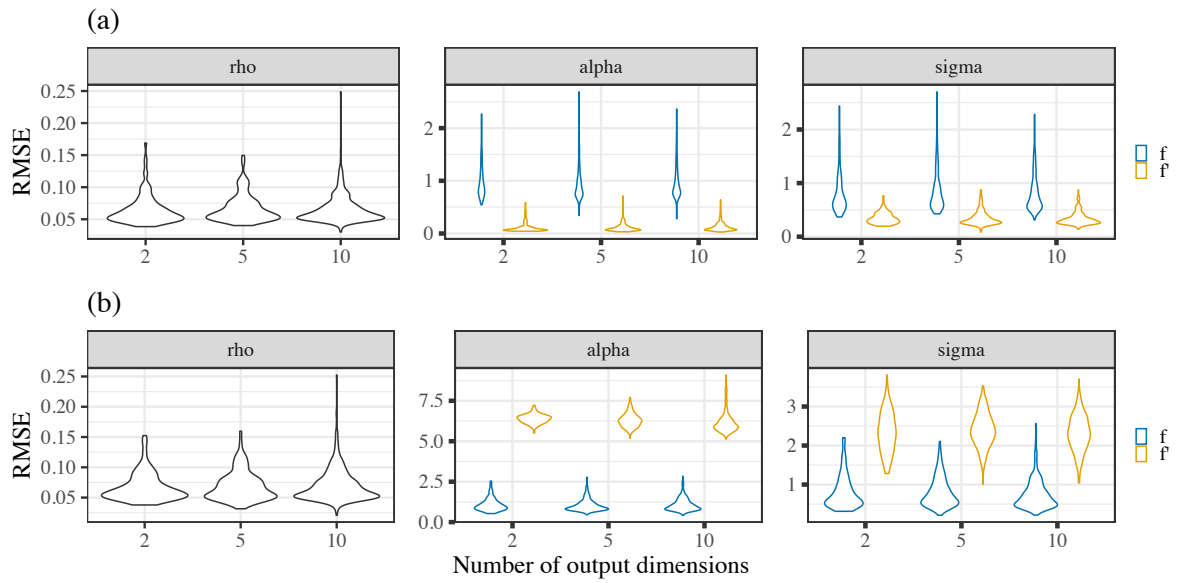

Figure 3.9: *Matérn 5/2 scenario: Hyperparameter RMSEs for (a) full DGP-LVM model and (b) models without scale assumption. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

are important when the underlying data require such complexities.

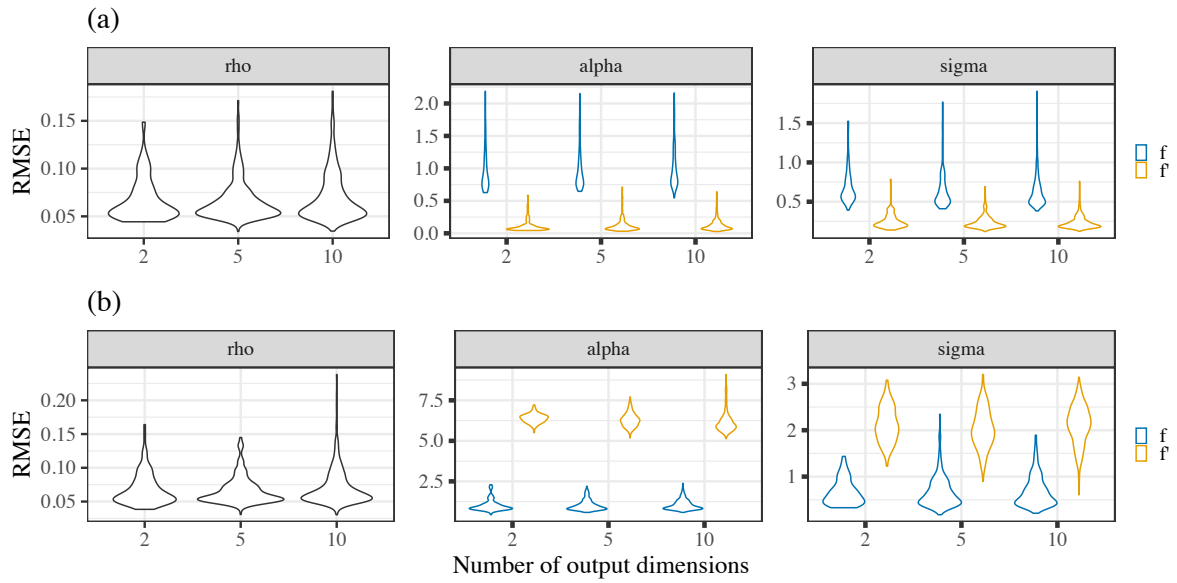

Figure 3.10: *Periodic scenario: Hyperparameter RMSEs for (a) full DGP-LVM model and (b) models without scale assumption. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

Special case: Non-stationary data

In the additional case of periodic data with trend, Figure 3.11 shows a sharp increase in RMSE values for higher dimensions (when $D = 10$). We believe this to be a direct consequence of modeling non-stationary data with stationary GPs. With more number of non-stationary output dimensions, more of the stationary GPs fail to model the data appropriately, thus

showing poor model performance in terms of recovering the ground truth of the latent $\mathbf{x}$. This special simulation scenario highlights one of the limitations of our current framework, which we further discuss in Section 3.6.1. Interestingly, for the $D = 10$, the problem vanishes when modeling correlated outputs. This is due to the fact that added trend behavior is same across outputs owing to the shared inputs, thus being highly correlated to one another. Due to this, accounting for correlated outputs seem to improve the recovery of true latent inputs.

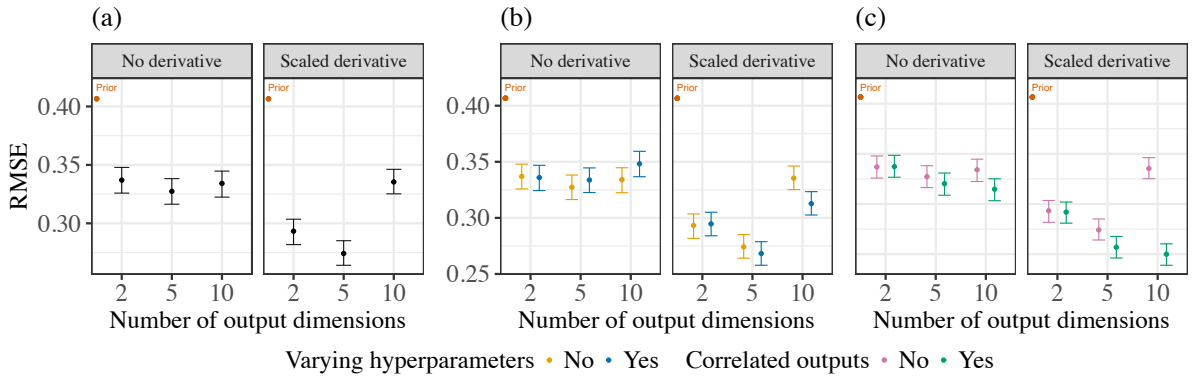

Figure 3.11: *Periodic with trend scenario: Main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs*

3.5 Case study

We showcase the application of DGP-LVM to real-world scRNA sequencing data by re-analyzing cell-cycle data from [56]. This dataset comprises of single-cell RNA expression profiles along the cell cycle as well as corresponding RNA velocities as estimates of expression profile derivatives obtained as a pre-processing step of cytopath, a method for simulation based cell trajectory inference [37]. Briefly, we choose this dataset because it covers single-cell transcriptomic profiles of the cell cycle, i.e. a cyclic process going through four phases depicting substantial variation in gene expressions and velocities.

For the purpose of this case study, we use a reduced data set of spliced RNA gene expression data and its corresponding RNA velocity of 20 cells and 12 genes. In other words, each sample point corresponds to a single cell and each output dimension corresponds to a single gene, with the value being the gene expressions per cell. Thus, for this case study, we have the sample points $N = 20$ for $\mathbf{y}$ and correspondingly $\mathbf{y}^{(1)}$ each with output dimensions $D = 12$. We sub-sampled the dataset in a stratified fashion so that cells from all four phases are included. We use the experimental time known as "cell hours" in the context of this specific data as the prior $\tilde{\mathbf{x}}$ for our latent pseudotime (input) $\mathbf{x}$. Both $\tilde{\mathbf{x}}$ and $\mathbf{x}$ are real numbers with values ranging between 0 and 1. For our prior measurement SD, we choose $s = 0.03$, so that it is proportional to our choices in simulation studies in Section 3.4.1.

We fit DGP-LVMs with derivative SE and Matérn 5/2 covariance functions. For this case study, We specify the priors for GP marginal SDs $\boldsymbol{\alpha} \sim \text{Normal}^{+}(13.84, 3.46^2)$ and $\boldsymbol{\alpha}^{(1)} \sim \text{Normal}^{+}(1, 0.25^2)$. In case of error SDs we specify $\boldsymbol{\sigma} \sim \text{Normal}^{+}(6.92, 3.46^2)$ and $\boldsymbol{\sigma}^{(1)} \sim$

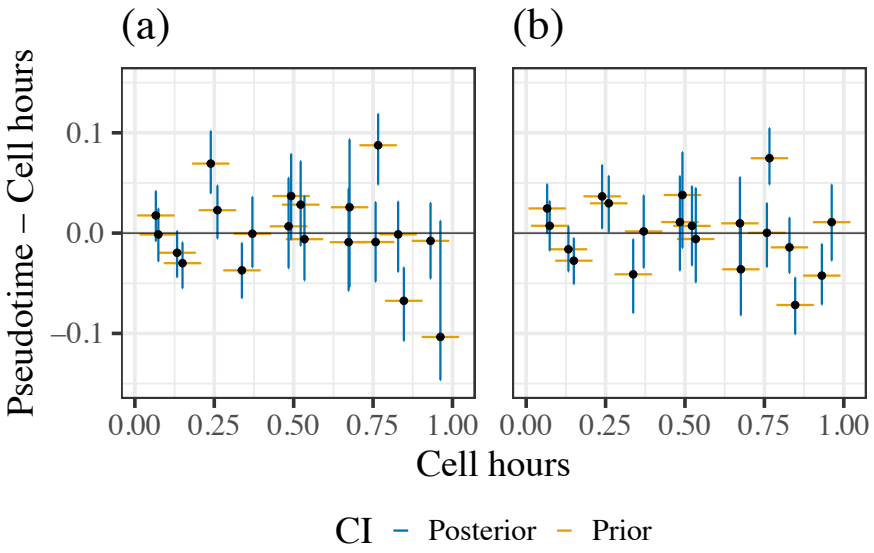

Figure 3.12: *Difference of latent pseudotime estimates obtained via DGP-LVM with (a) SE and (b) Matérn 5/2 covariance functions and prior cell hours. The point ranges horizontally show 95% prior CIs and vertically show 95% posterior CIs. Notice that the posterior CIs are actually much smaller than the prior CIs since y-axis scale is significantly smaller than x-axis.*

$\text{Normal}^+(0.5, 0.25^2)$. These choices were influenced by the large scaling factor λ informed by the data, as the mean and standard deviations of $\mathbf{y}$ are on average 13.84 times larger than those of $\mathbf{y}^{(1)}$ across the output dimensions. The prior for $\boldsymbol{\rho}$ was specified as $\text{Normal}^+(0.4, 0.1^2)$ for the SE and $\text{Normal}^+(0.6, 0.1^2)$ for the Matérn 5/2 as our GP length scale priors loosely based on the scale of the latent input $\mathbf{x}$ (as suggested by the prior $\tilde{\mathbf{x}}$). These priors were chosen to account for the varying functional smoothness induced by the choice of covariance function. The DGP-LVM with Matérn 3/2 being the least functionally smooth choice of covariance function from the Matérn family didn't converge reasonably with a sensible choice of $\boldsymbol{\rho}$ prior for this specific data and is therefore not presented.

As in any real-world latent variable estimation problem, we lack the ground truth to compare the estimated latent values against. Therefore, we study the deviation of the posterior estimates of pseudotime from the cell hours (our prior) by considering the difference or shift in values of the estimated pseudotime from the observed cell hours. The results are shown in Figure 3.12 with cell hours (prior) on the x-axis and shift (difference of pseudotime and cell hours) on the y-axis. Deviations from the $y = 0$ line indicate that latent pseudotimes are different from their cell hours (prior) as a result of learning from gene expression data and velocities. For some cells, prior-posterior differences are up to 5% of the total time scale. Further, we see that the posterior uncertainties (error bars in y-direction) are substantially smaller (considering the scale of the y-axis compared to x-axis) than the corresponding prior uncertainties (error bars in x-direction), which also indicates that model learning has taken place. Combined with our findings from the simulation study as strong evidence that DGP-LVM is able to learn and recover the true posterior estimates of the latent pseudotimes, these deviations from the prior are interpreted as model learning in the correct direction closer to the true latent ordering of

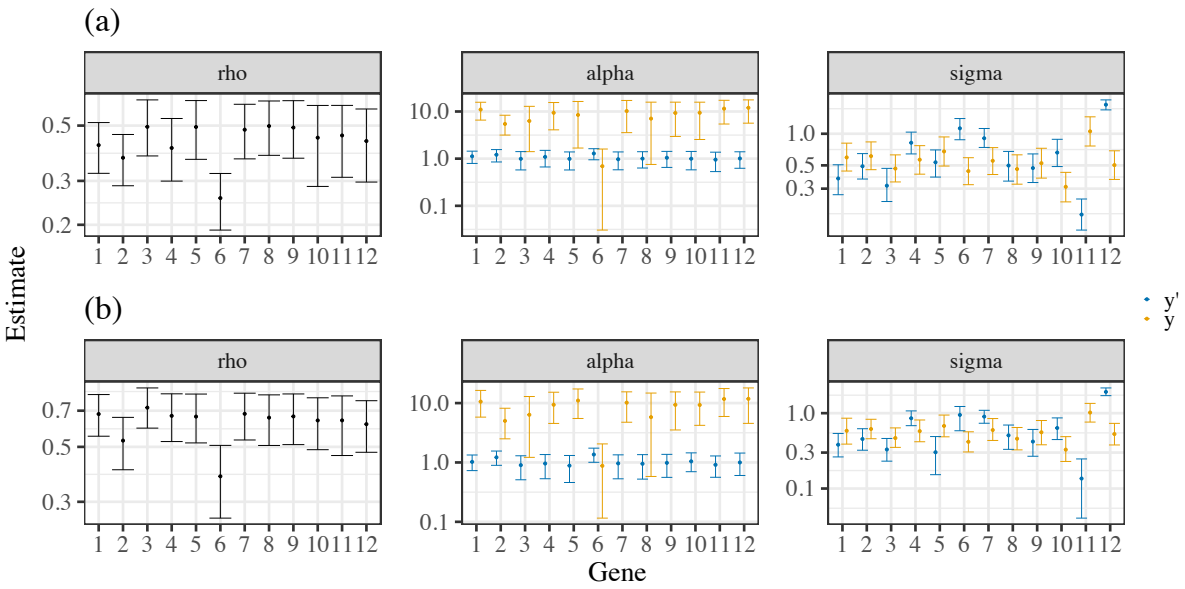

Figure 3.13: *Hyperparameters for DGP-LVM with (a) SE and (b) Matérn 5/2 covariance functions. The points indicate posterior mean and the point ranges indicate 95% CIs for each hyperparameter per output dimension. The different colors denote correspondence to the output* $\mathbf{y}$ *or its derivative* $\mathbf{y}^{(1)}$.

the cells.

In Figure 3.13, we show the posterior mean along with SD estimates of GP hyperparameters. We see strongly varying length-scales $\boldsymbol{\rho}$, marginal SDs $\boldsymbol{\alpha}$ and error SDs $\boldsymbol{\sigma}$ across different genes (output dimensions) for both SE and Matérn 5/2 models. This clearly points to the necessity of modeling hyperparameters as varying across genes. We also see substantial scale differences between the GP marginal SDs $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^{(1)}$ corresponding to $\mathbf{f}$ and $\mathbf{f}^{(1)}$, and consequently gene expression y and velocity $\mathbf{y}^{(1)}$ outputs, respectively. Similar, but not as drastic results are seen for the error SDs $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^{(1)}$. These results indicate significant scale differences between output RNA gene expression and its derivative RNA velocity. Interestingly, the scale differences go in both directions, such that for some genes $\boldsymbol{\alpha}$ and $\boldsymbol{\sigma}$ are higher than $\boldsymbol{\alpha}^{(1)}$ and $\boldsymbol{\sigma}^{(1)}$. For others, the direction is opposite. While the direction is not important for the DGP-LVM models, it may be highly relevant for understanding the biological processes in which the specific genes are involved. The difference in posterior estimates of the hyperparameters in the different models are heavily influenced by the natural varying functional smoothness corresponding to the choice of different covariance functions.

That said, this case study is meant only as a simple example for demonstrating the application of DGP-LVMs on real-world data. We would like to caution against any specific biological interpretation of the results at this point. The case study was conducted using Apple M1 chip (2 cores in parallel) with 16 GB memory allowances. The runtime (in minutes) for DGP-LVM were approximately 25.3 for the SE model and 73.11 for the Matérn 5/2 model.

3.6 Discussion

Motivated by a real-world problem in the area of single-cell biology, we developed a class of derivative Gaussian process latent variable models, DGP-LVMs. In the real-world case, we aim at estimating the latent ordering of cells from RNA gene expression levels and its corresponding time derivative RNA velocity. For this purpose, DGP-LVMs not only account for scale differences between the outputs and their derivatives, but also learn from multiple, potentially correlated outputs. The latter is highly important for scRNA sequencing data where the latent cell ordering is informed by many genes, each forming their own output coupled with derivative information.

In our simulation studies, we extensively validate DGP-LVMs demonstrating strong improvements in estimation accuracy of the latent variables by including derivative information. Our results also clearly show the importance of our proposed covariance function modifications. While we specifically focused on modifying the SE and Matérn class of covariance functions, our framework is generally applicable for any choice of covariance function that is twice differentiable.

3.6.1 Limitations and Future Research

This paper is only the first step towards tackling latent variable (input) estimation with derivative Gaussian processes. The current main limitation of DGP-LVMs is their data-scalability as they cannot be easily applied to large amounts of data, such as full sized scRNA sequencing datasets, yet. For a dataset of size N, exact GPs have a complexity of $O(N^3)$ in operations and $O(N^2)$ in memory. In case of multi-output GPs with correlated outputs and varying hyperparameters, the complexities increase to $O(N^3D + ND^2)$ and $O(N^2D + ND^2)$ respectively, where D is the number of output dimensions [40]. Additionally, when performing Bayesian inference via HMC involving a total of T unnormalized log posterior evaluations, the number of operations increases to even $O(N^3DT + ND^2T)$. Together, this limits inference for exact GPs on data with large N or D. From a real-world data point of view, scRNA sequencing data frequently has a few thousand cells (sample size N) with the number of genes (output dimensions D) being in the high hundreds after standard pre-processing steps. In case of DGP-LVM, this issue is even more severe due to adding derivative information, effectively doubling the sample size N. To address the computational limitations of DGP-LVM in terms of data-scalability, future research should consider extending approximate GP approaches [e.g., 72] to our DGP-LVM framework.

Another limitation of our current DGP-LVMs is their stationary assumption based on the choice of the covariance functions we discuss here. This limits their applicability to non-stationary data as evidenced by our simulation study of periodic data with an added non-linear trend. While this is a general limitation of stationary GPs, the limitation currently lies in not having a derivative version of non-stationary covariance functions. An interesting future research would be to develop DGP-LVMs for non-stationary data where the primary focus would be on obtaining derivative versions of non-stationary covariance functions and verifying their performance for latent variable estimation.

Another aspect for future research is the choice of prior distributions. Here, we focused on informative priors for the GP hyperparameters in both our simulation studies and the real-world case study although they are difficult to come by organically. DGP-LVMs will likely benefit from using stronger priors informed by the application-specific subject matter knowledge, specifically in data-sparse scenarios. This not only applies to priors for the GP hyperparameters, but also to the priors of the latent input variables. Moreover, a joint prior on the covariance function hyperparameters along with latent inputs will likely further improve model convergence. The combination of scalable approximations, improved prior specifications, and additional derivative covariance functions would foster the general applicability of DGP-LVMs, thereby further increasing their ability to accurately estimate latent variables.

Code availability

The code for the model development, simulation studies as well as the results can be found here: `https://github.com/Soham6298/DGP-LVM`.

Acknowledgments

We acknowledge the Cluster of Excellence iFIT (EXC 2180) "Image-Guided and Functionally Instructed Tumor Therapies" for supporting Soham Mukherjee. We acknowledge the valuable insights provided by members of ClaassenLab and BürknerLab, Revant Gupta and Marcello Zago. We thank Debapratim Sil and Jayati Chatterjee for their feedback on the manuscript. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Soham Mukherjee.

Chapter 4

Hilbert space methods for approximating multi-output latent variable Gaussian processes

Gaussian processes are a powerful class of non-linear models, but have limited applicability for larger datasets due to their high computational complexity. In such cases, approximate methods are required, for example, the recently developed class of Hilbert space Gaussian processes. They have been shown to drastically reduce computation time while retaining most of the favorable properties of exact Gaussian processes. However, Hilbert space approximations have so far only been developed for uni-dimensional outputs and manifest (known) inputs. To this end, we generalize Hilbert space methods to multi-output and latent input settings. Through extensive simulations, we show that the developed approximate Gaussian processes are indeed not only faster, but also provide similar or even better uncertainty calibration and accuracy of latent variable estimates compared to exact Gaussian processes. While not necessarily faster than alternative Gaussian process approximations, our new models provide better calibration and estimation accuracy, thus striking an excellent balance between trustworthiness and speed. We additionally validate our findings in a real world case study from single cell biology.

Declaration

This chapter is based on the following manuscript that is under review:

> *Hilbert space methods for approximating multi-output latent variable Gaussian processes*
> **Soham Mukherjee**, Manfred Claassen and Paul-Christian Bürkner
> arXiv preprint arXiv:2505.16919 – in review (2025)
> https://doi.org/10.48550/arXiv.2505.16919

Text, figures and tables are adapted from the manuscript https://arxiv.org/abs/2505.16919 with minor updates in notations.

Author contributions

Soham Mukherjee: Conceptualization, Methodology, Implementation, Validation, Formal Analysis, Writing – Original Draft, Writing – Review and Editing.

Manfred Claassen: Writing – Review and Editing, Supervision.

Paul-Christian Bürkner: Conceptualization, Writing – Review and Editing, Supervision.

4.1 Introduction

Gaussian processes (GPs) are a popular class of non-linear models well-known for their flexibility in modeling a wide range of data scenarios [94]. They are also well-known for their steep computational complexity, scaling cubically with the sample size. Thus, exact GPs cannot be used to model large datasets within feasible time. These scalability issues are even more pronounced for extensions of standard GP models. Here, we consider a combination of GPs for latent variable estimation [49, 50] and GPs with multi-dimensional outputs [25, 85]. Multi-output GPs are tasked to jointly model several, potentially correlated outcome variables, each with their own GP. In turn, estimating latent variables via GPs not only increases the number of estimable parameters drastically, but also requires multiple (potentially many) outputs to achieve sufficient estimation accuracy. Multi-output GPs for latent variable estimation have important applications, for example, in single-cell biology where estimating latent cell orderings from multi-dimensional RNA gene sequencing data is crucial to understand the dynamics of the biological process [39]. Yet, when using exact GPs for this purpose, only small datasets can be analyzed, which strongly limits their practical applicability.

Several methods have been proposed as scalable approximations for multi-output GPs [6, 40] and latent variable GPs [87, 88]. These methods primarily depend on reduced rank representations that approximate the GP covariance matrix using inducing points [67, 73, 76] followed by approximate model inference via mean-field variational inference (VI) ([87]. These GPs scale linearly with sample size and quadratically only with the number of inducing points. Thus, they are computationally efficient as long as the number of inducing points is not too large. However, the lack of support for custom prior specifications of GP hyperparameters limit the applicability to complex data scenarios. Moreover, an in-depth study of the statistical properties of latent variable estimates using these methods are, to the best of our knowledge, yet to be discussed.

An alternative approach is to approximate the covariance function through its spectral decomposition computed from a finite set of representative basis functions. The latter method falls under the category of Hilbert space approximate GPs [HSGPs; 72, 78]. By exploiting the spectral representation of a stationary covariance function, the computational complexity of HSGPs scales linearly with both sample size and the number of basis functions. What is more, they come with powerful diagnostics that indicate whether the chosen number of basis functions was sufficient for an accurate GP approximation given the data at hand [72]. HSGPs are typically estimated via Markov-chain Monte Carlo (MCMC), which is slower than VI but produces posterior approximations of much higher quality. The increased, diagnosable accuracy of HSGPs combined with their efficiency has a lot of potential for latent variable estimation. However, HSGPs have so far only been developed for single outputs and manifest (known) inputs. We develop extensions for the HSGP framework to address both latent variable inputs and multi-dimensional outputs and compare their benefits over exact GPs as well as other GP approximation methods.

4.1.1 Overview of Contributions

- We generalize HSGPs for latent variable inputs and multi-dimensional outputs.
- We validate our new HSGPs against the corresponding exact GPs (where feasible) as well as against variational GPs based on inducing points approximation.
- We perform extensive simulation studies to investigate the statistical properties of HSGPs in terms of latent variable estimation accuracy, uncertainty calibration, and model convergence.
- We demonstrate the applicability of our HSGPs on real-world single-cell RNA sequencing data, whose analysis is infeasible with similarly specified exact GPs.

4.2 Related work

Among the extensions of GPs, previous works discuss exact multi-output GPs [85] as well as their scalable counterparts [6, 40] for high-dimensional data. Recent works in multi-output GPs [43, 59] also account for output-dimension specific information using different GP hyperparameters for each output dimension. In case of latent variable GPs [49, 50] current approximations use a combination of inducing points [67, 76] and mean-field VI [87, 88] for scalable solutions. A combination of the above approximation methods for multi-output and latent variable GPs are presented in [48]. In this paper, we consider HSGPs [72, 78] and extend it for multi-dimensional outputs and latent variable inputs. Recently, [60] developed exact multi-output latent variable GPs with derivative information. We only take inspirations from the relevant parts (multi-output and latent inputs) of that work for our model development and experimental design.

In terms of real-world applications, we highlight the field of single-cell biology. There, multi-output structures naturally arise in a plethora of research problems [39] and latent GPs are common to model cellular ordering [20, 41]. Later works involve estimating the cellular ordering [21, 70] along with inference on different cellular pathway dynamics. A direct combination of the above multi-output latent GPs for single-cell RNA sequencing data is presented in [3], which used the combination of inducing points and VI methods for estimating cellular ordering. As a real-world case study, we showcase our models for a single-cell data of the cell cycle [56].

4.3 Methods

We initially discuss the structure of exact GPs and HSGPs for univariate output $\mathbf{y}$ and covariate $\mathbf{x}$ (known) with paired samples (y_i, x_i) for $i = 1, 2, ..., N$ where N is the sample size. Then, we generalize the Hilbert space framework for multi-dimensional outputs and latent inputs before discussing methods of inference and alternate GP approximation methods.

4.3.1 Gaussian processes

A Gaussian process (GP) is a stochastic process specified by a mean function $\mu = \mu(x)$ and a covariance function $k = k(x, x')$ where $x, x' \in \mathbb{R}$ are two arbitrary input values. We write

$\mathbf{f} \sim \mathcal{GP}(\mu, k)$ to indicate that $\mathbf{f} = \{f(x_1), \ldots, f(x_N)\}$ is distributed as a GP such that any finite subset of $\mathbf{f}$ jointly follows a multivariate Gaussian distribution. For a univariate output $\mathbf{y}$, a standard Gaussian process regression is given by

$$y_i = f(x_i) + \varepsilon_i \tag{4.1}$$

where ε_i is the i^{th} sample of the independent additive noise ε with error variance σ^2 such that $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. The entire model can thus be written as

$$y_i \mid f \sim \mathcal{N}(f(x_i), \sigma^2). \tag{4.2}$$

For $i \neq j$, $\text{Cov}(y_i, y_j) = k(x_i, x_j)$ such that $i, j = 1, 2, ..., N$. When $i = j$, $\text{Cov}(y_i, y_j) = \text{Var}(y_i) = k(x_i, x_j) + \sigma^2$. In terms of k, the Matérn class of stationary covariance functions is a highly common choice [69, 94]. Among them, we specifically consider Squared Exponential (SE), Matérn 3/2 and 5/2, which are defined as:

$$\begin{aligned} k_{\text{se}} &= \alpha^2 \exp\left(-\frac{r^2}{2\rho^2}\right), \\ k_{\text{m32}} &= \alpha^2 \left(1 + \frac{\sqrt{3r^2}}{\rho}\right) \exp\left(-\frac{\sqrt{3r^2}}{\rho}\right), \\ k_{\text{m52}} &= \alpha^2 \left(1 + \frac{\sqrt{5r^2}}{\rho} + \frac{5r^2}{3\rho^2}\right) \exp\left(-\frac{\sqrt{5r^2}}{\rho}\right), \end{aligned} \tag{4.3}$$

where $r = |x - x'|$ is the distance between two arbitrary inputs, $\alpha > 0$ is the marginal standard deviation (SD) and $\rho > 0$ is the length scale parameter. The above covariance functions vary in terms of their implied functional smoothness, with the SE covariance function being the most and the Matérn 3/2 being the least smooth. We use a constant mean function μ for all of our exact GPs and HSGPs discussed in this paper.

4.3.2 Hilbert space approximations

In the HSGP framework [72, 78], stationary covariance functions are approximated using their spectral densities in the frequency domain. Briefly, Bochner's theorem [31] states that a covariance function of a stationary process can be expressed as the Fourier transform of a positive finite measure. If that measure has a density, then we call it a spectral density of the covariance function. Further, using Wiener-Khintchine theorem [23], we can show that a covariance function and its spectral density are Fourier duals. Combining the results from both these theorems, any arbitrary function satisfying the criteria of being a covariance function (symmetric and positive-definite) can be expressed in terms of its spectral density. The spectral densities

corresponding to the covariance functions from Eq.(4.3) are given by

$$\begin{aligned} S_{\text{se}}(\omega) &= \alpha^2 \rho(\sqrt{2\pi}) \exp\left(-\frac{1}{2}\rho^2\omega^2\right) \\ S_{\text{m32}}(\omega) &= \alpha^2 \left(\frac{2\Gamma(2)3^{3/2}}{\rho^3/2}\right) \exp\left(\frac{3}{\rho^2} + \omega^2\right)^{-2} \\ S_{\text{m52}}(\omega) &= \alpha^2 \left(\frac{2\Gamma(3)5^{5/2}}{3\rho^5/4}\right) \exp\left(\frac{5}{\rho^2} + \omega^2\right)^{-3}, \end{aligned} \tag{4.4}$$

where $\omega \in \mathbb{R}$ is the input in the frequency domain, and $\rho > 0$ and $\alpha > 0$ are the covariance function hyperparameters. The spectral densities can be derived for the general Matérn class of covariance functions [72], but for this paper, we specifically focus on the above special cases.

We follow the procedure of [72] and start by defining the compact cover $\Omega \in [-L, L] \subset \mathbb{R}$ containing the vector of inputs $\mathbf{x}$ for any positive real number L. Following this, we can write any stationary covariance function with $x, x' \in \Omega$ as

$$k_\theta(x, x') = \sum_{j=1}^{\infty} S_\theta(\sqrt{\lambda_j})\phi_j(x)\phi_j(x'), \tag{4.5}$$

where S_θ is the spectral density of the covariance function k_θ with θ being the set of hyperparameters. The sets of eigenvalues $\{\lambda_j\}_{j=1}^{\infty}$ and eigenfunctions $\{\phi_j\}_{j=1}^{\infty}$ of the Laplacian operator in the domain Ω satisfy the following problem in Ω under Dirichlet boundary conditions [78]:

$$\begin{aligned} -\nabla^2\phi_j(x) &= \lambda_j\phi_j(x), \quad x \in \Omega \\ \phi_j(x) &= 0, \quad x \notin \Omega \end{aligned} \tag{4.6}$$

where the eigenvalues $\lambda_j > 0$ are real positive due to the Laplacian being a positive-definite Hermitian operator. Here, the ϕ_j are sinusoidal functions and the solution to the eigenvalue problem in Eq.(4.6) are independent of the choice of covariance functions. They are given by

$$\begin{aligned} \lambda_j &= \left(\frac{j\pi}{2L}\right)^2, \\ \phi_j(x) &= \sqrt{\frac{1}{L}} \sin\left(\sqrt{\lambda_j}(x + L)\right). \end{aligned} \tag{4.7}$$

We approximate the covariance function using the linear combination of its basis functions. Considering the first M number of basis terms, we obtain from Eq.(4.5),

$$k_\theta(x, x') \approx \sum_{j=1}^{M} S_\theta(\sqrt{\lambda_j})\phi_j(x)\phi_j(x'). \tag{4.8}$$

The full covariance matrix $\mathbf{K}$ for inputs x_i, $i \in 1, ..., N$ is thus approximated by its finite basis function as

$$\mathbf{K} \approx \mathbf{\Phi}\mathbf{\Delta}\mathbf{\Phi}^T \tag{4.9}$$

where $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$ is the matrix of eigenfunctions

$$\mathbf{\Phi} = \Phi(\mathbf{x}) = \begin{bmatrix} \phi_1(x_1) & \dots & \phi_M(x_1) \\ \vdots & \ddots & \vdots \\ \phi_1(x_n) & \dots & \phi_M(x_n) \end{bmatrix}. \tag{4.10}$$

for a vector of input points $\mathbf{x} = (x_1, \dots, x_n)$. We can thus re-write the GP functions from Section 4.3.1 as

$$\mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{\Phi \Delta \Phi}^T). \tag{4.11}$$

This is equivalent to the linear representation

$$f(\mathbf{x}) \approx \mu + \sum_{j=1}^{M} \left(S_\theta(\sqrt{\lambda_j}) \right)^{1/2} \Phi(\mathbf{x}) \beta_j \tag{4.12}$$

where $\beta_j \sim \mathcal{N}(0, 1)$ is standard normal. The eigenvalues λ_j are monotonically increasing in j and S_θ rapidly decreases to zero for a bounded covariance functions. As a result, usually only a small number of basis functions M are required for an adequate approximation [69, 72].

4.3.3 Extending HSGPs

We extend the HSGP framework to model multi-dimensional outputs and latent variable inputs. We follow a similar framework pertaining to the multi-output latent variable models considered in [60] (but do not consider the derivative methods here).

Multi-output HSGPs

We specify multi-output GPs with response variables $(\mathbf{y}_1, \dots, \mathbf{y}_D)$ over $D > 1$ output dimensions [69], using y_{di} to denote the response for dimension d and observation i. As the usual approach, we first set up D independent, univariate Gaussian processes $\mathbf{f}_d$ each with their own set of hyperparameters, say $\boldsymbol{\theta}_d$ [60, 85]. Thus, extending HSGPs to multi-output GPs, we first modify Eq.(4.12) such that

$$f_d(\mathbf{x}) \approx \sum_{j=1}^{M} \left(S_{\boldsymbol{\theta}_d}(\sqrt{\lambda_j}) \right)^{1/2} \Phi(\mathbf{x}) \beta_{jd} \tag{4.13}$$

where $\beta_{jd} \sim \mathcal{N}(0, 1)$. The univariate GPs are then related to one another by folding them with a (D-dimensional) across-dimension correlation matrix $\mathbf{C}$ [12, 85]. Specifically, for each observation i, we obtain a vector of across-dimension correlated GP values as

$$\begin{pmatrix} f_1^*(x_i) \\ \dots \\ f_D^*(x_i) \end{pmatrix} = A \times \begin{pmatrix} f_1(x_i) \\ \dots \\ f_D(x_i) \end{pmatrix}, \tag{4.14}$$

where A is the Cholesky factor of $\mathbf{C}$ such that $\mathbf{C} = \mathbf{A}\mathbf{A}^T$ with $\mathbf{A}$ being lower-triangular. This way, multi-output GPs combine two dependency structures, one within dimensions (and across observations) as expressed by the univariate GPs through corresponding covariance functions

and one across output dimensions (but within observations) as expressed by $\mathbf{C}$ (or $\mathbf{A}$). Adding independent Gaussian noise to our derivative multi-output GP model, we extend Eq.(4.2) for all d and i:

$$y_{di} \mid f_d^* \sim \mathcal{N}(f_d^*(x_i), \sigma_d^2) \tag{4.15}$$

An alternative way of specifying $\mathbf{C}$ is through a set of input-dependent covariance functions [13, 32]. In our study, we however only consider an input-independent $\mathbf{C}$, as modeling the across-dimension correlations is not the main focus of this paper. In Section 4.6, we provide a more detailed discussion on this topic.

Latent variable HSGPs

Within latent-variable GPs, the inputs $\mathbf{x}$ are considered as unobserved and are treated similar to other estimable parameters. To that end, we specify an observed quantity $\tilde{\mathbf{x}}$ that guides the estimation of the latent inputs $\mathbf{x}$. From a Bayesian perspective, $\tilde{\mathbf{x}}$ acts as a prior-like object for the latent $\mathbf{x}$ to be then further refined by the GPs learning from $\mathbf{y}$. Specifically, we consider the scenario where $\tilde{\mathbf{x}}$ is a noisy measurement of $\mathbf{x}$. If we assume that the measurements $\tilde{\mathbf{x}}$ are Gaussian with known measurement SD s, we can write for each observation i:

$$\tilde{x}_i \sim \mathcal{N}(x_i, s^2). \tag{4.16}$$

The vector of latent inputs $\mathbf{x} = (x_i, \ldots, x_n)$ is then passed to the approximation step in Eq.(4.12) (and subsequently Eq.(4.13) for the multi-output case). We call the resulting model (multi-output) latent variable HSGPs.

Latent variable GPs, exact or approximate, are more difficult to fit than their manifest counterparts. The primary reasons are the substantial increase in the number of estimable parameters as well as identification issues arising due to both $\mathbf{x}$ and $\boldsymbol{\rho}$ now being treated as unknown. We alleviate this identifiability issue using MCMC methods discussed in Section 4.3.4. We further present a detailed simulation study investigating the challenges of latent variable GPs in Section 4.4.

4.3.4 Bayesian inference

We fit HSGPs using full Bayesian inference via MCMC sampling. Our primary parameters of interest are the latent inputs $\mathbf{x}$ and GP hyperparameters $\boldsymbol{\theta}$ given data (outputs) $\mathbf{y}$. We re-purpose $\boldsymbol{\theta}$ to include not only the covariance function hyperparameters $\boldsymbol{\rho}$ and $\boldsymbol{\alpha}$ as specified earlier but also the error SD $\boldsymbol{\sigma}$. For a specific output dimension d, we assume independent prior distributions on $\boldsymbol{\theta}_d$ as

$$\boldsymbol{\theta}_d \sim p(\boldsymbol{\theta}_d) = p(\rho_d)\, p(\alpha_d)\, p(\sigma_d). \tag{4.17}$$

The joint probability density factorizes as

$$p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta} \mid \tilde{\mathbf{x}}) = p(\mathbf{x} \mid \tilde{\mathbf{x}}) \prod_d^D p(\mathbf{y}_d \mid \mathbf{x}, \boldsymbol{\theta}_d)\; p(\boldsymbol{\theta}_d). \tag{4.18}$$

where $p(\mathbf{y}_d \mid \mathbf{x}, \boldsymbol{\theta}_d \mid \tilde{\mathbf{x}})$ denotes the GP-based likelihood for a single output dimension and $p(\mathbf{x})$ denotes the prior for the latent $\mathbf{x}$ implied by the measurement model in Eq.(4.16). The details of prior specifications used in our experiments are further discussed in Section 4.4. Using Bayes' rule, we obtain the joint posterior over $\mathbf{x}$ and $\boldsymbol{\theta}$ as

$$p(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}, \tilde{\mathbf{x}}) = \frac{p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta} \mid \tilde{\mathbf{x}})}{\int \int p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta} \mid \tilde{\mathbf{x}})\, d\mathbf{x}\, d\boldsymbol{\theta}}. \tag{4.19}$$

The inference framework specified above is same for both exact and HSGPs. Posterior samples of $\mathbf{x}$ and $\boldsymbol{\theta}$ for all output dimensions are obtained via MCMC sampling, specifically adaptive Hamiltonian Monte Carlo [42, 62]. We implemented the exact GP and the HSGP models in Stan using the RStan interface [80].

4.3.5 Approximate latent variable GPs using VI

Approximating latent variable GPs using VI methods (VIGPs) usually include a combination of approximating both the covariance function and the posterior. Approximating the covariance function is carried out using inducing points to obtain a reduced rank representation of the Gram matrix [69]. There are many inducing point methods [67], however, based on the suggestions of [8], we specifically consider the implementation discussed in [87]. In this framework, the estimation of latent variables $\mathbf{x}$ and GP hyperparameters $\boldsymbol{\theta}$ after marginalizing over $\mathbf{f}$ is carried out via the marginal likelihood

$$p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{y} \mid \mu, \mathbf{K}_{N\times N} + \sigma^2\mathbf{I}), \tag{4.20}$$

where $\mathbf{K}_{N\times N}$ is the full rank GP covariance matrix for a sample size N. This full rank covariance matrix is then replaced by $\mathbf{Q}_{N\times N} = \mathbf{K}_{N\times M}\mathbf{K}_{M\times M}^{-1}\mathbf{K}_{M\times N}$ where M is the number of inducing points. This reduces the computational complexity of inverting a full rank matrix to only computing its Nyström approximation. While fundamentally different, the basis points in our method explained in Section 4.3.2 and inducing points here serve a similar purpose. For that reason, we choose to denote both the number of basis points (in HSGPs) and inducing points (in VIGPs) by M for simplicity. Effectively, the marginal likelihood is then reformulated as

$$p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}^*) = \mathcal{N}(\mathbf{y} \mid \mu, \mathbf{Q}_{N\times N} + \sigma^2\mathbf{I}). \tag{4.21}$$

Following the specifications discussed in [88], the joint probability distribution for the d^{th} output dimension is given by

$$p(\mathbf{y}_d, \mathbf{f}_d, M_d) = p(\mathbf{y}_d \mid \mathbf{f}_d)\, p(\mathbf{f}_d \mid M_d)\, p(M_d), \tag{4.22}$$

where M_d is the dimension-specific set of inducing points. The posterior $p(\mathbf{f}_d, M_d \mid \mathbf{y}_d) = p(\mathbf{f}_d \mid M_d, \mathbf{y}_d)\, p(M_d \mid \mathbf{y}_d)$ is subsequently approximated by a sparse variational distribution of the form

$$q_p(\mathbf{f}_d, M_d) = p(\mathbf{f}_d \mid M_d)\, q_M(M_d), \tag{4.23}$$

where $q_M()$ is the variational distribution over the inducing points M_d. Model inference is carried out by the usual method of computing the variational lower bound (see [88] for further

details). While the latent variable GPs were earlier designed for classification tasks ([49], [50]), VIGPs have later been used for GP regressions as well, thus making it an ideal candidate for reference and comparison to our proposed HSGPs. In this paper, we follow the implementations for VIGPs in [3, 88] using the GP-LVM module of Pyro [10].

4.4 Simulation Study

For real-world data, we lack the ground truth values of latent variables. In contrast, in a simulated setting, we have full control over the data generating process including the true latent values. Thus, it is crucial to investigate the statistical properties of latent variable models using simulated data. To this end, we investigate exact GPs, HSGPs and VIGPs under various simulation scenarios exploring their behavior in terms of model convergence, model calibration, and estimation accuracy. The data generating conditions are inspired by [60].

4.4.1 Data generating scenarios

We consider six different simulation scenarios. These scenarios are broadly categorized into data generated from a GP and data generated from a periodic function (non-GP data scenario). Under the former category, we consider three data generating processes based on multi-output GPs with SE, Matérn 3/2, and Matérn 5/2 covariance functions, respectively. We sample length-scale $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.05^2)$, GP marginal SD $\boldsymbol{\alpha} \sim \text{Normal}^+(3, 0.25^2)$, and error SD $\boldsymbol{\sigma} \sim \text{Normal}^+(1, 0.25^2)$. Additionally, we consider a fourth GP data scenario, where we allow the length-scale of the SE covariance function to be highly variable across each output dimensions, $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.25^2)$, thus increasing the challenges in model fitting. Each of the hyperparameters are sampled D times, one value per output dimension. Under the GP data scenarios, the fitted models align with the true data generating process (or are approximations to them). However, simulating from GPs results in a lot of variation in the true functional forms across output dimensions and simulation trials.

Compared to that, the periodic data generating process have more consistency since we ensure a similar level of non-linearity across each outputs. Concretely, we use the following periodic process:

$$\begin{aligned} f_{id} &= \alpha_d^2 \sin\left(\frac{x_i}{\rho_d}\right), \\ y_{id} &\sim \mathcal{N}(f_{id}, \sigma_d^2). \end{aligned} \tag{4.24}$$

In our simulations, we consider two version of this process, one with low oscillations (high true ρ_d values) and one with higher oscillations (low true ρ_d values). These two periodic data scenarios mimic the challenges of modeling starkly different functional smoothness with the higher oscillations being the more challenging one. The hyperparameters of the periodic process closely resemble that of the GP data scenarios and are thus named the same for simplicity. The sampling distributions are also similarly specified as mentioned previously. For the periodic data with higher oscillation, we specify $\boldsymbol{\rho} \sim \text{Normal}^+(0.5, 0.05^2)$ as compared to the lower oscillation case where $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.25^2)$.

In all scenarios, we sample the between-dimension (output) correlation matrix $\mathbf{C} \sim \text{LKJ}(\eta = 1)$ [52]. This implies $\mathbf{C}$ to be uniformly distributed within the set of all valid correlation matrices of dimension D. For the sample size N, we consider three cases $N = 20, 50$ and 200. We generate the ground truth latent inputs as $x_i \sim \text{Uniform}(0, 10)$, where $i = 1, \ldots, N$. Further, we assume a prior measurement SD of the noisy $\tilde{x}$ as $s = 0.3$ (see Section 4.3.3). For the number of output dimensions D, we consider the three cases $D = 5, 10$ and 20. We perform 50 trials for each of the choices of N and D per simulation scenario.

4.4.2 Model specifications

We fit exact GPs, HSGPs, and VIGPs for all simulation scenarios, with the exception of omitting exact GPs for sample sizes higher than $N = 20$ due to the scalability issues discussed earlier (see Section 4.1). For the HSGPs, we select the boundary conditions L by multiplying a scaler adjustment $c = 1.25$ to the range of the input prior $\tilde{\mathbf{x}}$. This way, we ensure that we have a compact cover $[-L, L]$ that prevents input values $\mathbf{x}$ to be near the boundaries. By doing so, we avoid model convergence issues completely (see Section 4.4.3). We select the minimum number of basis functions M_{min} loosely based on the suggestions of [72]. The choices for M_{min} depends on the choice of covariance function to accurately approximate the degree of non-linearity. Thus, for the SE covariance function,

$$M_{\text{min}} = 1.75\frac{cS}{\mu_\rho}, \tag{4.25}$$

for the Matérn 3/2,

$$M_{\text{min}} = 3.42\frac{cS}{\mu_\rho}, \tag{4.26}$$

and for the Matérn 5/2,

$$M_{\text{min}} = 2.65\frac{cS}{\mu_\rho}. \tag{4.27}$$

In the above equations, c is the scaler adjustment to the boundary conditions, S is the range of inputs and μ_ρ is the mean of the length-scale prior. The SE covariance function requires much lower M_{min} compared to, say, the Matérn 3/2 which is on the other end of the smoothness spectrum among the Matérn class.

Prior specifications for the exact GPs and HSGPs are kept same for all model parameters. Our prior choices are aligned with the data generating process since that is required for testing model calibration (see Section 4.4.4). Thus, we specify priors for length-scale $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.05^2)$, GP marginal SD $\boldsymbol{\alpha} \sim \text{Normal}^+(3, 0.25^2)$ and error SD $\boldsymbol{\sigma} \sim \text{Normal}^+(1, 0.25^2)$. For the periodic data with higher oscillation, we specify a lower prior mean of $\boldsymbol{\rho} \sim \text{Normal}^+(0.5, 0.05^2)$. Lastly, in case of the SE simulation scenario with highly varying length-scale, we specify a prior of $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.25^2)$.

Following the suggestions of the number of inducing points discussed in [87], we use $M = 10$ inducing points for the VIGPs. In this framework, a small constant is added to the diagonal terms of the matrix. This helps maintain numerical stability while computing the GP covariance matrix. In case of VIGPs, we tested different values of diagonal constants starting from $\delta = 10^{-12}$ (usually suggested default). Based on our experience and data generating conditions, the

VIGPs need $\delta = 10^{-3}$ for the GP covariance matrices to reliably become numerically positive definite. Higher number of inducing points would require an even larger δ. Since $\delta = 10^{-3}$ is already quite large, we keep $M = 10$ to not further increase the bias caused by large values of δ. As in the other methods, we set the prior measurement SD to $s = 0.3$. The VIGP implementation in Pyro does not support priors on the GP hyperparameters, so we kept them at their uniform defaults.

Both exact GPs and HSGPs are implemented in Stan [80] and fitted with a single MCMC chain of 2000 iterations of which the first 1000 are discarded as warm-up. As shown in [60], model convergence is similar for multiple chains when fitting latent variable GPs. Thus, we run only a single chain per model to reduce overall computation times. The simulation studies are conducted on 50 vCPUs (Intel(R) Xeon(R) Gold 6230R CPU @ 2.10 GHz) with 720 GB work memory. The average runtime per dataset for the HSGPs with $N = 200$ and $D = 20$, the most computationally expensive case, was approximately 28 minutes for the SE model, 42 minutes for the Matérn 3/2 model, and 34 minutes for the Matérn 5/2 model. In the same scenarios, the VIGPs took approximately 2 minutes regardless of the covariance function. The exact GPs (just for $N = 20$ and $D = 20$) took 48 minutes for the SE model, 1.6 hours for the Matérn 3/2 and 2.4 hours for the Matérn 5/2 model. The HSGPs were significantly faster than the exact GPs but slower than VIGPs. The slower speed of HSGPs compared to VIGPs is counter-balanced by highly consistent model calibrations, superior estimation accuracy for latent variables, and overall much more stable model performance, as shown below.

4.4.3 Model convergence

We investigate MCMC convergence of our fitted exact GPs and HSGPs for all of the six simulation study scenarios discussed before. We use standard MCMC sampling diagnostics including state-of-the-art versions of the scale reduction factor $\widehat{R}$, bulk effective sample size (Bulk-ESS) and tail effective sample size (Tail-ESS) [91]. A combined check of these measures provide a comprehensive picture of the parameter-specific model convergence. In general, $\widehat{R}$ should be very close to 1 and should ideally not exceed 1.01 [91]. We additionally consider a more relaxed threshold of 1.1 in our simulation studies, since the ground truth is available as another layer of evaluation. Bulk-ESS indicates the reliability of measures of central tendency such as the posterior mean or median. Tail-ESS indicates the reliability of the 5% and 95% quantile estimates, which are then used to construct credible intervals. Both Bulk-ESS and Tail-ESS should have values greater than 100 times the number of MCMC chains (higher is better). We compute all the convergence measures with the posterior package [19]. Inference of the VIGPs is done via numerical optimization, which we considered to have converged if no convergence warnings were issued by the optimizer. In Figure 4.1, we show $\widehat{R}$, Bulk-ESS, and Tail-ESS for the latent $\mathbf{x}$ and GP hyperparameters of both exact GPs and HSGPs with SE covariance function. While the exact GPs reach and exceed the relaxed $\widehat{R}$ threshold of 1.1 for some simulated datasets, the HSGPs consistently satisfy the much stricter 1.01 threshold of model convergence. HSGPs subsequently also have much higher Bulk and Tail-ESS. Overall, based on the diagnostics, HSGPs show much more consistent and stable convergence as compared to exact GPs. We provide

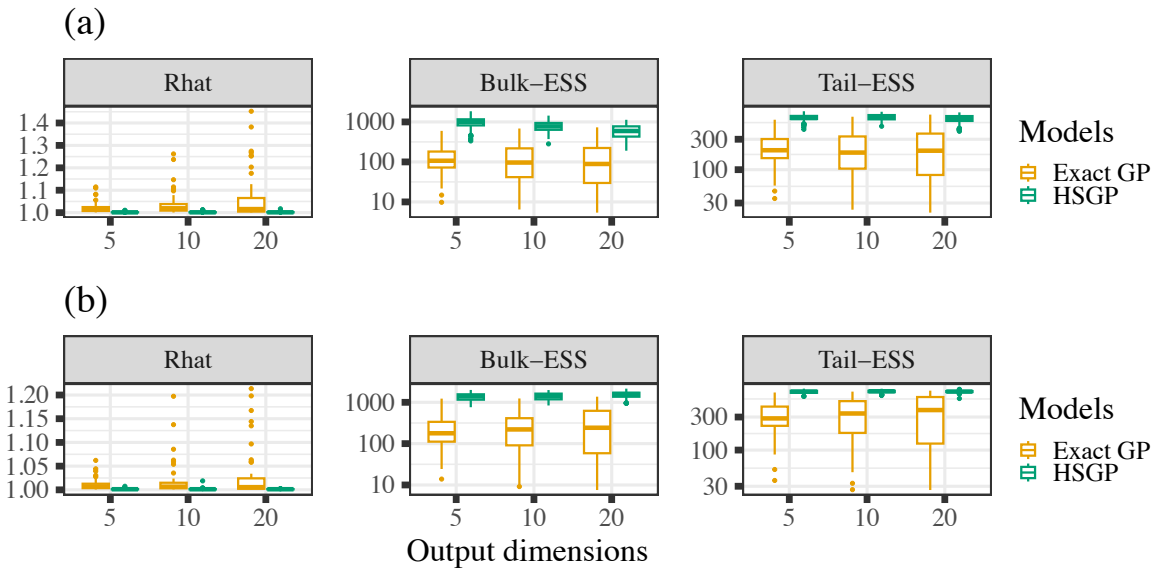

Figure 4.1: *Squared exponential scenario: Convergence check for (a) latent inputs and (b) GP hyper-parameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

convergence diagnostic figures for the other simulation scenarios in the Appendix B.3 (Figures B3-B7) where we see similar results.

4.4.4 Testing model calibration

We use simulation based calibration (SBC) [58, 84] to test model calibration for estimating latent inputs $\mathbf{x}$. Through SBC, we check if the model is able to produce posterior distributions that are consistent with the data generating process, thus correctly accounting for the implied uncertainty. To that end, briefly, SBC aims to test the goodness of posterior approximations

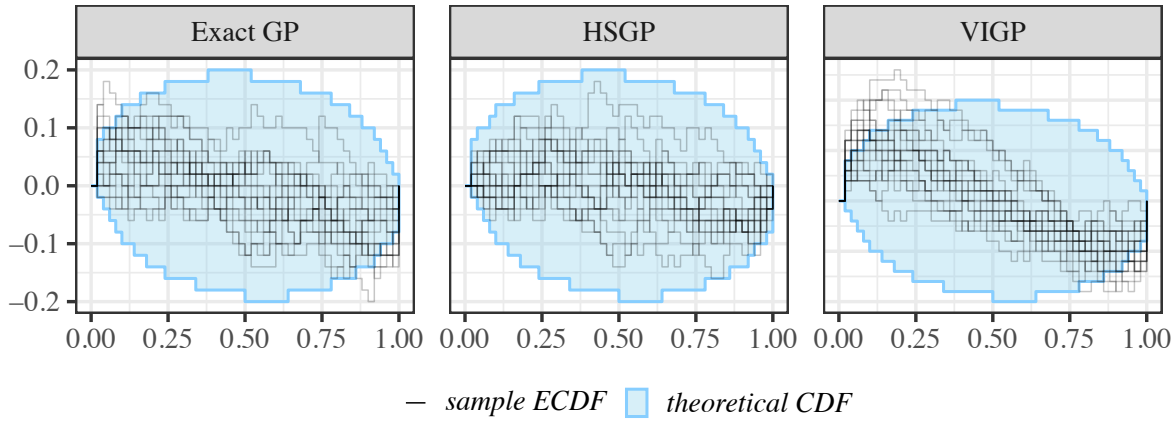

Figure 4.2: *Squared exponential scenario: ECDF-difference calibration plots of the latent* $\mathbf{x}$ *estimated by exact GP, HSGP, and VIGP. Only the HSGP is consistently well calibrated.*

by exploiting self-consistency properties of Bayesian models. This results in a rank statistic that is uniformly distributed under the assumption of a well calibrated model [58], which can subsequently be checked both graphically and numerically. As a graphical test, we plot the

empirical cumulative distribution function (ECDF) of the ranks along with their 95% confidence regions under the assumption of uniformity [83]. ECDFs lying outside of their confidence region indicate miscalibrations. These graphical tests also yield a test statistic [the log γ score; see 83], which we further analyze numerically. More details on SBC is provided in Appendix B.1.

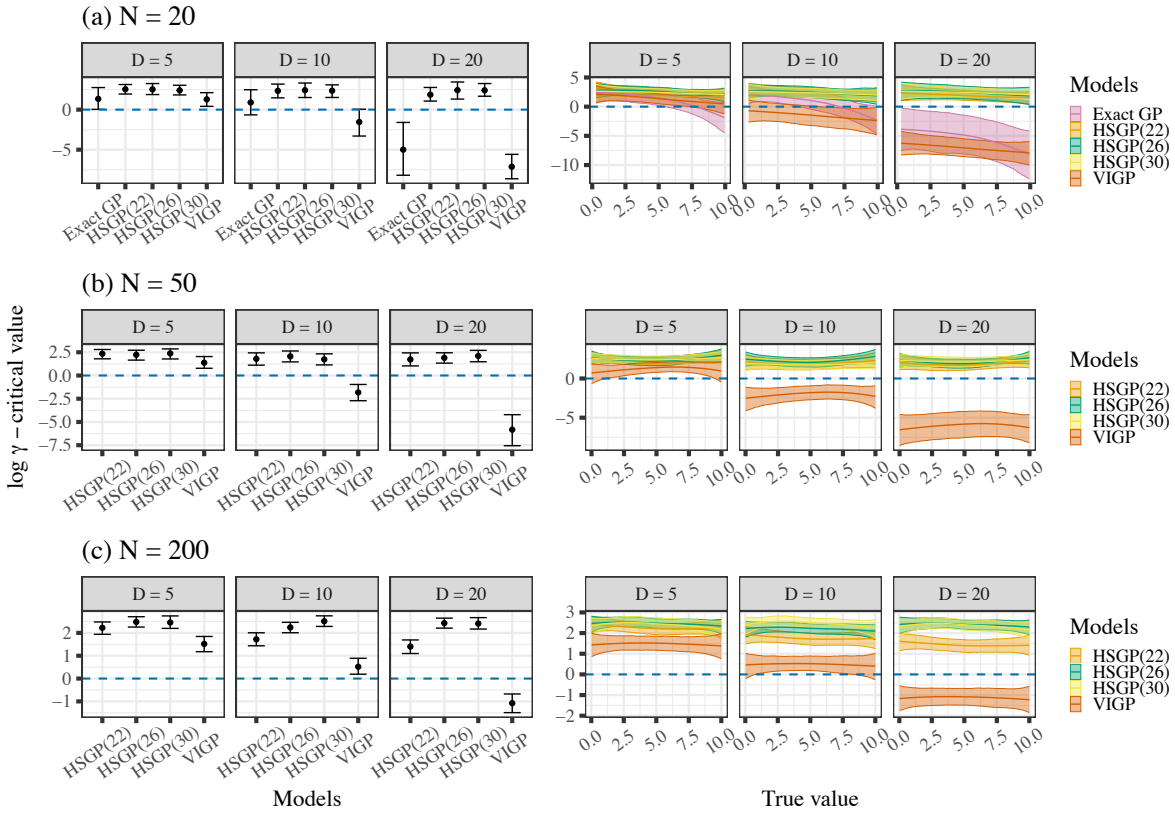

Figure 4.3: *Squared exponential scenario: log γ scores offset by the 95% confidence threshold for all the fitted models. The behavior of scores across true latent* $\mathbf{x}$ *values are shown in the right-hand panel. The blue dashed line denotes the threshold to reject uniformity. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

In Figure 4.2, we show the rank ECDFs obtained from the exact GPs, HSGPs, and VIGPs for the SE simulation scenario ($N = 20$ and $D = 20$). The results indicate that the latent variable estimates obtained from the HSGPs are consistently well calibrated. In contrast, for exact GPs, we see miscalibrations for some latent variables shown by their corresponding rank sample ECDFs lying outside the threshold region. In case of VIGPs, the instances of miscalibrations are even more severe.

To streamline calibration checking across simulation scenarios, we analyze the log γ scores using a multi-level model [17] described in Appendix B.1. In Figure 4.3, we show the predicted log γ scores of the different GP models under the SE simulation scenario. For all the choices of N and D, the latent variable estimates from HSGPs are consistently well calibrated. By comparison, the exact GPs fail the calibration test for $D = 20$. The VIGPs shows miscalibrations especially for higher output dimension cases throughout all the choices of N. Additionally, exact GPs exhibit worse calibrations (decreasing log γ scores) when estimating larger values of the latent inputs. This issue directly causes problems in estimating latent values closer to the boundary.

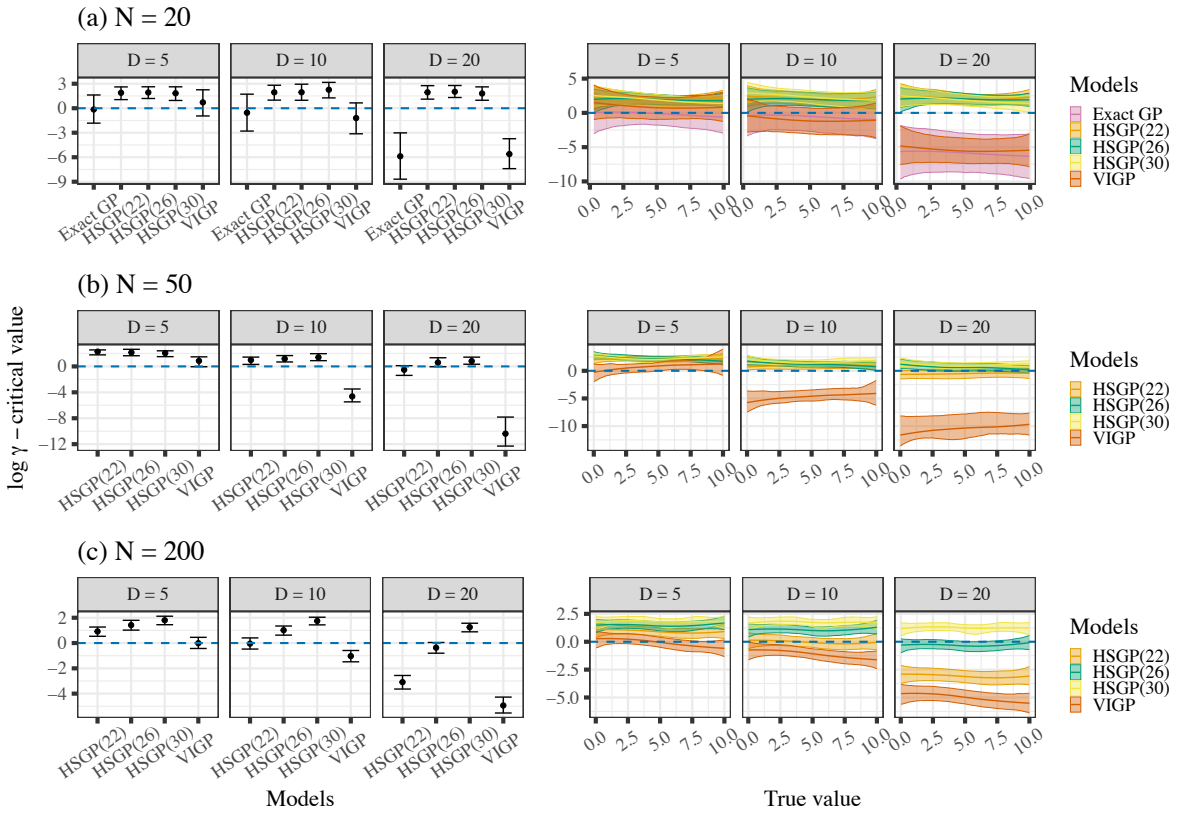

Figure 4.4: *Squared exponential scenario (highly varying ρ): log γ scores offset by the 95% confidence threshold for all the fitted models. The behavior of scores across true latent* $\mathbf{x}$ *values are shown in the right-hand panel. The blue dashed line denotes the threshold to reject uniformity. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

VIGPs show similar behavior for lower sample sizes $N = 20$ although not as strong as exact GPs. The HSGPs are able to overcome and rectify this behavior. For the special scenario of SE data with highly variable length-scales as shown in Figure 4.4, VIGPs and HSGPs with small number of basis function struggle to pass the calibration tests, especially for $N = 200$. The miscalibrations are however alleviated with a higher number of basis functions in HSGPs. The log γ results for the Matérn 3/2 and 5/2 simulation scenarios are presented in Appendix B.4 (Figures B8 and B9).

4.4.5 Latent variable estimation

For all of the simulation scenarios discussed in Section 4.4.1, we evaluate the estimation capabilities of the different fitted models specified in Section 4.4.2. We show model performance in terms of absolute bias $|\text{Bias}(\mathbf{x}_{post}, \mathbf{x}_{true})|$ of the latent posterior estimates $\mathbf{x}_{post}$ with respect to their true values $\mathbf{x}_{true}$ as well as $\text{SD}(\mathbf{x}_{post})$ as a measure of posterior sharpness. Since mean-field VI is known to underestimate the posterior SD [46], we avoid using a metric like RMSE that favors (too) narrow posteriors, thus providing misleading interpretations if used as a sole metric. We summarize our findings using the multi-level model [17] described in Appendix B.1 for both posterior bias and SD. The SE simulation scenario presented in Figure 4.5 shows that, in most cases, HSGPs have lower bias compared to exact GPs and VIGPs. The VIGPs

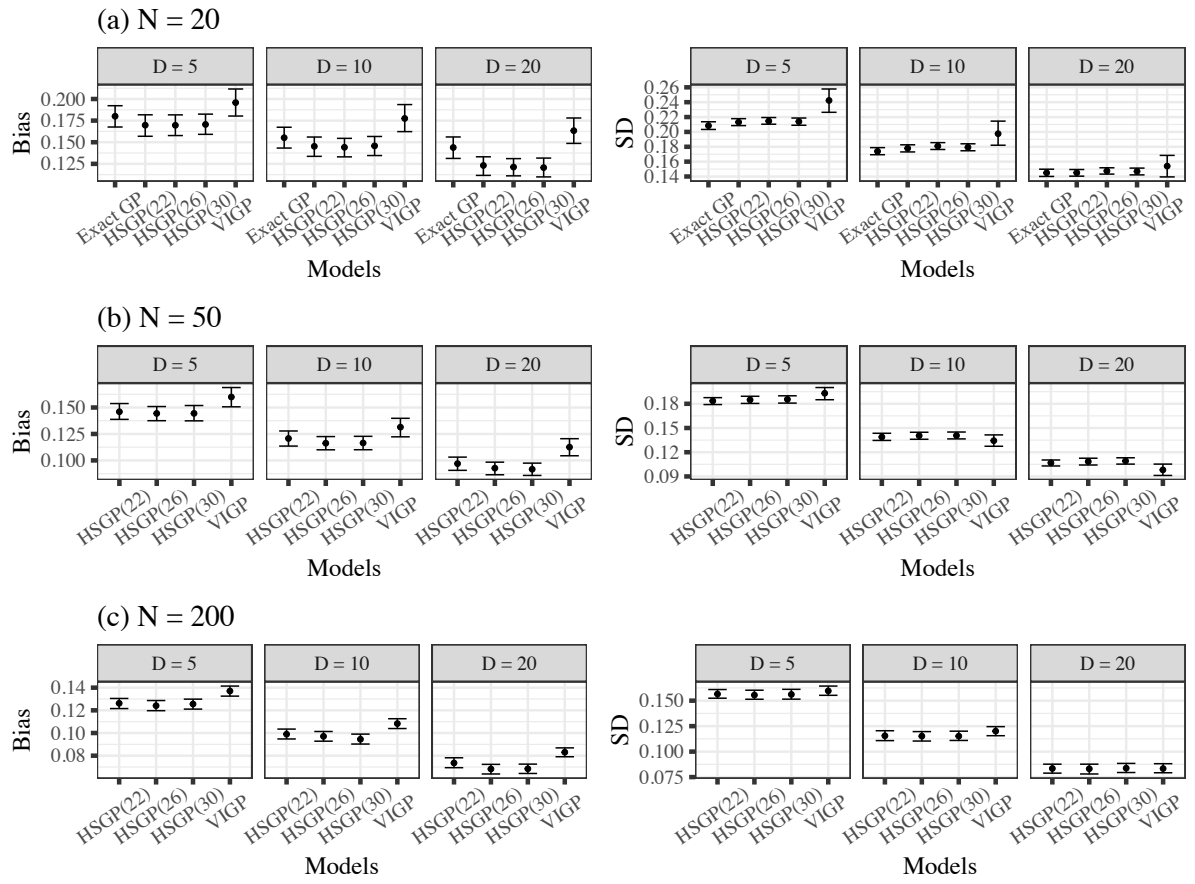

Figure 4.5: *Squared exponential scenario: posterior bias and SD on recovery of latent inputs for all fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

consistently show the highest bias. Overall, for the SE data scenario, we see on average about 11% (for higher sample sizes) to about 18% (for lower sample sizes) decrease in posterior bias between VIGPs and HSGPs (with $M = 22$ basis functions). Results for the posterior SD are more mixed, with VIGPs showing higher SDs for small sample sizes and about the same SDs for larger sample sizes.

In Figure 4.6 we show the more challenging simulation scenario. Here, the data generating process had a higher amount of variability in length-scales across the output dimensions. We observe a stronger difference in posterior bias between VIGPs and HSGPs (in favor of HSGPs). We hypothesize that HSGPs are more strongly favored here as they allow for varying hyperparameters across dimensions, while the VIGPs (at least their Pyro implementation) only support constant hyperparameters.

Among the HSGPs, higher number of basis functions ($M = 26$ and 30 compared to $M = 22$) result in lower posterior bias for both the simulation scenarios shown in Figures 4.5 and 4.6. Combining this result with their corresponding model calibration tests, we recommend using $M = 30$ to achieve the highest accuracy for the latent variable estimates. For faster but reliable results, HSGPs with $M = 22$ can be used in simpler cases like the SE data scenario. Furthermore, in the remaining scenarios (see Figures B10-B13 in Appendix B.5), HSGPs perform

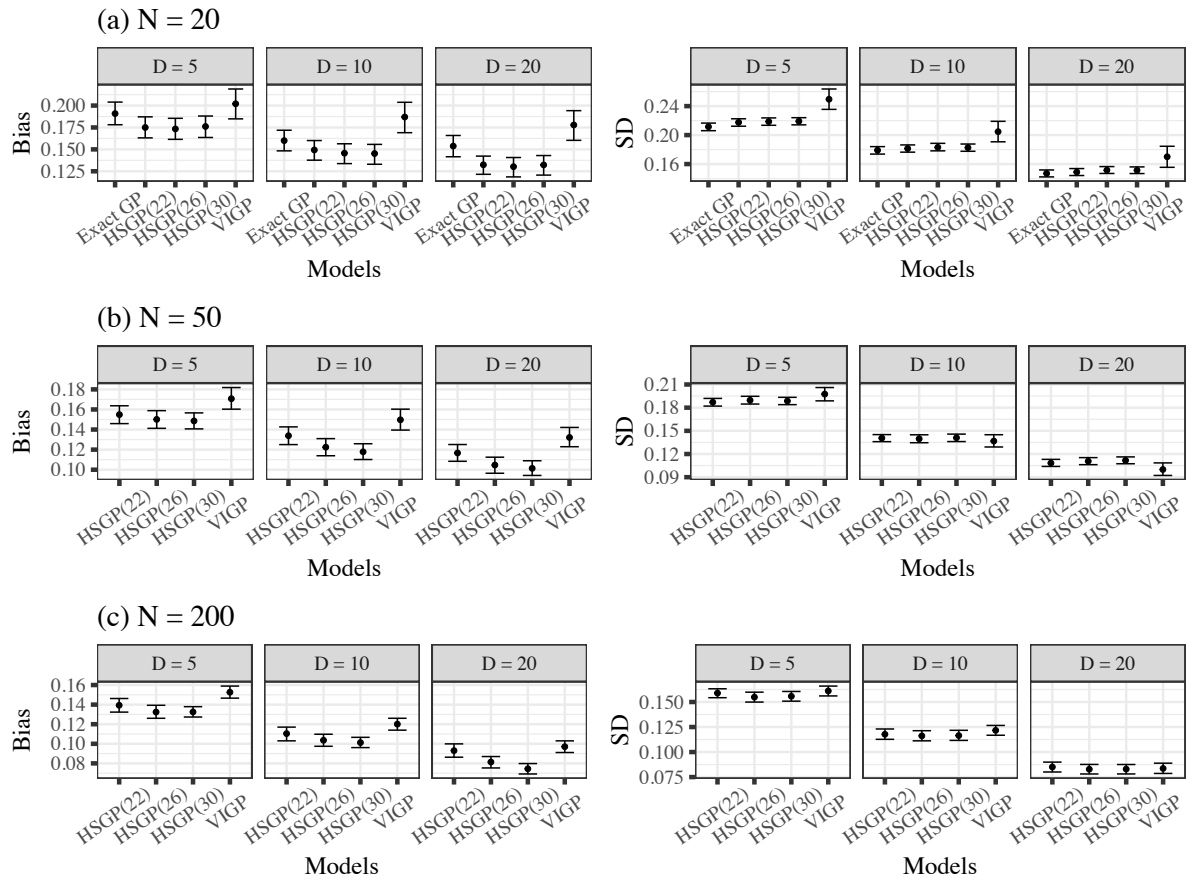

Figure 4.6: *Squared exponential scenario (highly varying ρ): posterior bias and SD on recovery of latent inputs for all fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

consistently better than their competitors. In all the scenarios, each of the fitted models (exact GPs only for $N = 20$) show a consistent decrease in posterior bias and SD as the number of output dimensions D and sample size N increases. We report the estimation results for GP hyperparameters in the Appendix B.2.

4.5 Real-World Case Study

We showcase the application of the HSGPs to real-world scRNA sequencing data by re-analyzing cell-cycle data from [56]. Our aim is to estimate the underlying cell ordering known as "pseudo-time". The dataset comprises of spliced single-cell RNA expression profiles along the cell cycle with their own time sequence for cells that are aligned with the experimental time called "cell hours". We choose this dataset because it covers a cyclic process going through three distinct biological phases (G1, G2M and S-ph) depicting substantial variation in gene expressions. Additionally, the provided cell hours is a continuous temporal sequence that can be utilized as a prior guide to estimate pseudotime. For the purpose of this case study, we use the data from all the cells along with a selection of 12 influential genes (see Appendix B.7 for the list) based on the recommendations of [56]. In other words, each sample point corresponds to a single cell and each output dimension corresponds to a single gene, with the value being the gene

expressions per cell. Thus, the analyzed data comprises of $N = 960$ observations and $D = 12$ output dimensions. We use cell hours in the context of this specific data as $\tilde{\mathbf{x}}$ for our latent

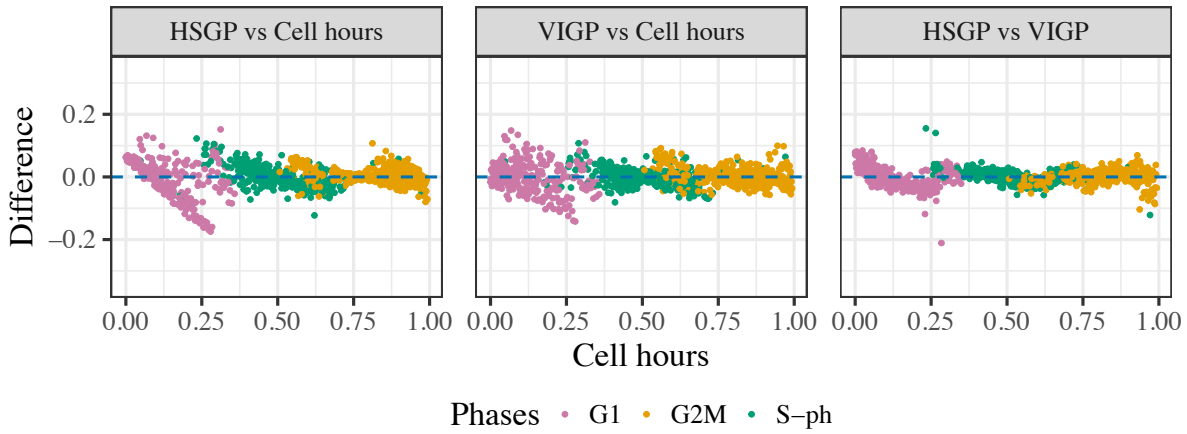

Figure 4.7: *(a) Difference of HSGP pseudotime estimates from the prior cell hours; (b) Difference of VIGP pseudotime estimates from the prior cell hours; (c) Difference of HSGP and VIGP pseudotime estimates. Both models assume an SE covariance function.*

pseudotime $\mathbf{x}$. Both $\tilde{\mathbf{x}}$ and $\mathbf{x}$ are real numbers with values ranging between 0 and 1. For our prior measurement SD which is unknown, we choose $s = 0.03$, keeping it proportional to our choices in the simulation studies in Section 4.4.1 and 4.4.2.

We fit a HSGP and VIGP with SE covariance function. We specify the priors for length-scale $\boldsymbol{\rho} \sim \text{Normal}^{+}(0.4, 0.1^2)$, GP marginal SD $\boldsymbol{\alpha} \sim \text{Normal}^{+}(14, 3.5^2)$, and error SD $\boldsymbol{\sigma} \sim \text{Normal}^{+}(7, 3.5^2)$. The prior specifications are based on the similar case study in [60]. Based on the $\boldsymbol{\rho}$ prior and the $(0, 1)$ range of the input space, we use the minimum number of basis function $M = 6$ (see Equation 4.25). For the VIGP, we again resort to default priors since custom priors are not supported. Similarly, we use 10 inducing points as the reasonable choice without compromising the numerical stability.

As in any real-world latent variable problem, we lack knowledge of the ground truth for the latent variable. Thus, we show the deviation of the posterior pseudotime $\mathbf{x}$ estimates from the prior cell hours $\tilde{\mathbf{x}}$. The results from our simulation studies provide evidence that this shift is indeed in the direction of the ground truth. The pseudotime deviations from cell hours for both HSGP and VIGP are presented in Figure 4.7. The first two sub-figures show the estimated pseudotime deviations from the HSGP and VIGP, respectively, plotted against the prior cell hours. The third sub-figure show the difference between the estimated pseudotimes from these two models. For some cells, prior-posterior differences are on average about 10% of the total cell hours time scale for both HSGP and VIGP. Comparing their two pseudotime estimates, we see differences of about 5% (with some outlying cells).

In Figure 4.8, we show the posterior estimates of the HSGP hyperparameters along with their posterior 95% CIs. A key observation here is the strong differences in the hyperparameter estimates across the genes (output dimensions). This confirms how crucial it is for the model to assume gene-specific hyperparameters while modeling these biological processes. Modeling

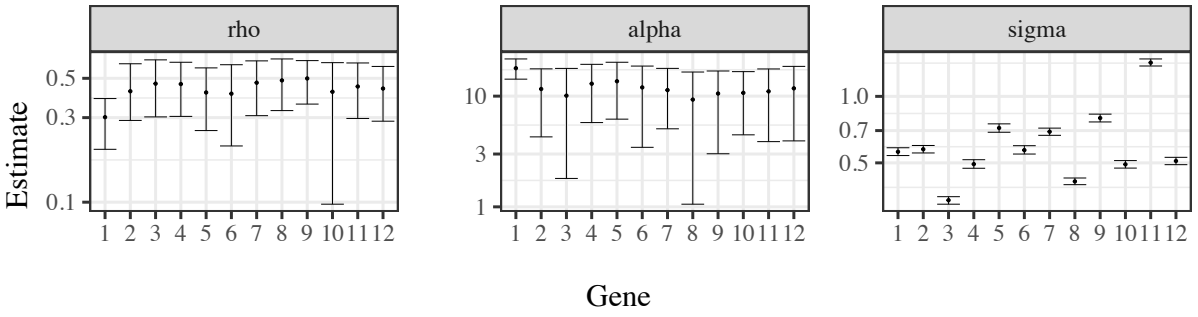

Figure 4.8: *Hyperparameter estimates for the HSGP with SE covariance function for cell cycle data. Points indicate posterior means and the point ranges indicate 95% CIs for each hyperparameter per output dimension.*

varying GP hyperparameters, while supported in our HSGPs, is not possible yet for the (pyro implemented) VIGPs, so we do not include their estimates in Figure 4.8.

The runtime for the HSGP was approximately 16 minutes on an Apple M4 chip (2 cores in parallel) with 24 GB working memory. Fitting the VIGP took about 3 minutes using the cluster computing resources detailed in Section 4.4.2.

4.6 Discussion

We generalize Hilbert space approximations for Gaussian processes [HSGPs; 72, 78] to develop a scalable class of latent variable models for high-dimensional multi-output data. Our HSGPs account for both varying hyperparameters across output dimensions and correlations between output dimensions. They reduce the steep computational complexity of $O(N^3D + ND^2)$ for exact multi-output GPs to $O(NMD + ND^2)$ for a dataset of N samples and D output dimensions, where M is the number of basis functions of the HSGP. Since M is typically much smaller than N, using HSGPs reduce computation times drastically. Through extensive simulations, we investigated the statistical properties of latent GPs in terms of model convergence, uncertainty calibration, and estimation accuracy. In all of the above criteria, the HSGPs performed better than both exact GPs and approximate GPs based on inducing points and variational inference (VIGPs)[3, 87]. While VIGPs are faster yet, they provide limited modeling capabilities and are outperformed by HSGPs in terms of model calibration and estimation accuracy. Even though HSGPs are comparably slower, they only need a few minutes even for relatively large datasets such as the one investigated in our real-world case study. This stands in sharp contrast to exact GPs, which would have taken at least several hours, if not more, to fit the same data.

4.6.1 Limitations and future research

For GPs with Matérn class of covariance functions, the length-scale $\boldsymbol{\rho}$ is directly related to the scale of the input space. In latent variable GPs, this implies strong dependencies between the prior on $\boldsymbol{\rho}$, the range of latent $\mathbf{x}$, and the prior measurement SD s (see Sections 4.3 and 4.4.2). As a result, specifying all three independently may lead to convergence issues and overall bad model behavior. Future research should investigate these dependencies more closely to provide

better guidelines for prior specification in latent variable GPs.

With regard to the correlation structure $\mathbf{C}$ across output dimensions (see Section 4.3.3), we make the choice of modeling it on the level of the GP functions $\mathbf{f}$ rather than on the level of observations $\mathbf{y}$. It is not always apparent which of these two options is preferable for a given application. For example, in spatial statistics, correlations between discrete locations are also either modeled on the mean (similar to the GP functions in our case) level, e.g., via conditional autoregressive (CAR) structures, or on the observation level, e.g., via spatial autoregressive (SAR) structures [92]. In this paper, we apply our methods to RNA gene expressions for cell cycle data, where the genes are known to be correlated at the biological process level [71]. Thus, specifying across-dimension correlations on the GP functions appear sensible here. For other applications, it may be preferable to specify the correlations at the observational level, or even jointly at both levels.

Another consideration concerning $\mathbf{C}$ is whether to model it as input dependent or input independent. In this paper, we chose it to be input independent, thus estimating the full correlation matrix directly. As an alternative, an input dependent formulation could be achieved through another GP that models the correlations across output dimensions within each observation (similar to a cross-covariance structure shown in [32]). In this regard, it would further be interesting to study Hilbert space approximations also for this across-output dimension GP, especially when the number of dimensions is large enough for exact GPs to become computationally impractical. We leave these extensions of our framework to future research.

Lastly, an important generalization of latent variable GPs concern the inclusion of derivative information, which can further increase estimation accuracy [60]. However, such methods have so far only been developed for exact GPs, which hamper their real-world applicability. Extending HSGPs to include derivative information will likely provide a scalable solution. However, such an extension requires spectral densities for composite covariance functions, which to our knowledge is yet to be derived.

Code availability

The code for the model development, simulation studies as well as the results can be found here: `https://github.com/Soham6298/Latent-variable-HSGPs`

Acknowledgments

We acknowledge the valuable insights provided by all group members. P.B further acknowledges support of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via the Collaborative Research Center 391 (Spatio-Temporal Statistics for the Transition of Energy and Transport) – 520388526. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting S.M.

Chapter 5

Latent variable estimation with composite Hilbert space Gaussian processes

We develop a scalable class of models for latent variable estimation using composite Gaussian processes, with a focus on derivative Gaussian processes. We jointly model multiple data sources as outputs to improve the accuracy of latent variable inference under a single probabilistic framework. Similarly specified exact Gaussian processes scale poorly with large datasets. To overcome this, we extend the recently developed Hilbert space approximation methods for Gaussian processes to obtain a reduced-rank representation of the composite covariance function through its spectral decomposition. Specifically, we derive and analyze the spectral decomposition of derivative covariance functions and further study their properties theoretically. Using these spectral decompositions, our methods easily scale up to data scenarios involving thousands of samples. We validate our methods in terms of latent variable estimation accuracy, uncertainty calibration, and inference speed across diverse simulation scenarios. Finally, using a real world case study from single-cell biology, we demonstrate the potential of our models in estimating latent cellular ordering given gene expression levels, thus enhancing our understanding of the underlying biological process.

Declaration

This chapter is based on the following manuscript that is under review:

> *Latent variable estimation with composite Hilbert space Gaussian processes*
> **Soham Mukherjee**, Javier Enrique Aguilar, Marcello Zago, Manfred Claassen and Paul-Christian Bürkner
> arXiv preprint arXiv:2510.25371 – in review (2025)
> https://doi.org/10.48550/arXiv.2510.25371

Text, figures and tables are adapted from the manuscript https://arxiv.org/abs/2510.25371 with minor updates.

Author contributions

Soham Mukherjee: Conceptualization, Methodology, Implementation, Validation, Formal Analysis, Writing – Original Draft, Writing – Review and Editing.

Javier Enrique Aguilar: Methodology, Writing – Review and Editing.

Marcello Zago: Formal Analysis, Writing – Review and Editing.

Manfred Claassen: Writing – Review and Editing, Supervision.

Paul-Christian Bürkner: Writing – Review and Editing, Supervision.

5.1 Introduction

Gaussian processes (GPs) are a powerful class of non-parametric methods, allowing to tackle a wide range of research problems [94]. Latent variable estimation, a major branch in statistical modeling, is among these problems solvable with GPs and its extensions [69]. Latent variable GPs have not only been used for classification and dimension reduction [5, 49, 50], but also for regression models [11]. Other extensions of GPs include multi-output GPs [12, 85] and derivative GPs [77]. Multi-output GPs, as the name suggests, use multi-dimensional (rather than uni-dimensional) data as their joint output. Derivative GPs, on the other hand, simultaneously model derivative information in addition to the original data through a joint derivative covariance function structure. Derivative GPs are a special case of jointly modeling two GPs with shared inputs. We will refer to this general framework as *composite GPs*. While the naming of this general structure as composite GPs aren't popular (to the best of our knowledge), early uses of composite GPs can be found in, for example, specifying the joint predictive distribution of GPs [69].

A commonality between both the multi-output and derivative GP extensions is to simultaneously model various sources of data in a single GP framework. Leveraging this aspect, [60] recently combined all of the aforementioned extensions for the purpose of latent variable estimation with multi-output derivative GPs, leading to drastic increase in accuracy for latent variable estimates. However, a major drawback of this framework lies in its applicability to large data, since exact GPs scale cubically with the number of observations. In this paper, we propose a scalable version of the composite GPs for latent variable estimation using multi-dimensional outputs that overcomes the scalability issues through the use of approximate GPs. As a special case, we will also show scalable derivative GPs designed for the same purpose.

Various types of approximation methods for GPs have been discussed over the years. A major direction among them is to consider a reduced-rank representation of the Gram matrix based on the chosen covariance function [67, 73, 75, 76]. In these works, the reduced rank representation is achieved by selecting a small set of representative samples known as inducing points. By solving the covariance function only on the inducing points, computation becomes much faster compared to inverting the Gram matrix of the full sample.

Another line of research under the reduced-rank representation approach focuses on Hilbert space approximate GPs (HSGPs) to speed up GP computation [72, 78]. Under the HSGP framework, the covariance function is approximated through its spectral decomposition which is computed from a finite set of representative basis functions. Exploiting the spectral representation of a stationary covariance function, HSGPs scale linearly with both sample size and the number of basis functions. Recently, HSGPs were extended to multi-output latent variable GPs in [61] where the authors demonstrated their advantages in fast GP computation and estimating well-calibrated latent variables. Here, we further generalize the framework to composite GPs, with a specific focus on derivative GPs.

As an application of our developed framework, we tackle a contemporary research problem in single-cell biology where we estimate latent cellular ordering from single-cell RNA (scRNA)

gene expression data [39]. This latent cellular ordering provides a proxy for the properties of the underlying biological processes [90] and is thus a crucial component that is to be estimated. To achieve this, we model unspliced and spliced gene expression data using composite GPs. Recently, latent cellular ordering were estimated based on RNA velocity [9, 47], which is obtained by estimating the rate of change between the spliced and unspliced gene expressions. Thus, we provide an alternative approach to estimating latent cellular ordering using spliced gene expression and RNA velocity through a scalable derivative GP model. ScRNA gene expression data is known for their large number of samples (cells) along with several inter-correlated genes. Concretely, we show how our proposed framework overcomes the aforementioned scalability issues and models full-sized scRNA gene expression data with thousands of cells, thus providing a practical approach of accurately estimating latent cellular ordering.

5.1.1 Overview of contributions

We develop a scalable class of composite GPs and its special case, derivative GPs for latent variable estimation. Based on our theoretical results, we provide a way to easily obtain spectral densities for a general derivative covariance function. We utilize these spectral densities to obtain an Hilbert space approximation for derivative GPs, as well as composite GPs more generally. Through extensive simulations, we show that our methods achieve accurate latent variable estimation scaling up to large data scenarios that are impractical for exact GPs. We demonstrate the potential of our methods in estimating latent cellular ordering from full-sized single-cell gene expression data, thus enhancing our understanding of the underlying biological process.

5.2 Related work

In the field of latent variable modeling, GPs constitute a broad class of models. Latent variable GPs were introduced in [49, 50] tasked towards dimensionality reduction via point estimation methods. Later works introduced estimating latent variable inputs using Bayesian inference [88] as well as approximate variational inference methods for applications to high-dimensional data scenario [40]. Recently, a generalized latent variable estimation framework supporting various inference strategies were presented in [48]. Other GP extensions include modeling multi-dimensional outputs [12, 85] and simultaneous modeling of derivative information alongside GP outputs [77]. Recently, all of the three above extensions were combined in a single modeling framework [60], for which we propose a scalable version, among other contributions made in this paper.

While GPs are lauded for their flexible modeling capabilities, practical applications remain prohibitive for large sample sizes due to the computational requirements for exact GPs. Strategies like inducing points approximations [67, 75, 76] combined with fast variational inference [8, 87] have been developed to address the practical limitations of GPs. Recently, Hilbert space GPs (HSGPs) [72, 78] were developed where the covariance function is approximated through its spectral decomposition for a scalable solution. Furthermore, the multi-output latent variable HSGPs [61] promise well-calibrated, fast inference for latent variable estimates from large data

scenarios. Here, we further generalize the HSGP framework to composite GPs, with a specific focus on derivative GPs [60].

GPs have been popularly used in the field of single-cell biology [15, 16, 41]. Specifically, various works estimate the latent cellular ordering [21, 70] along with branching structures for trajectory inference [3] using GPs for understanding the true biological process. A limitation in these works lies in their restricted use of analyzing gene expression levels from a single source of information. In this paper, we showcase how incorporating additional information on gene expression levels results in an increased accuracy for these latent ordering estimates through our developed models.

5.3 Methods

We generalize the HSGP framework to composite and derivative GPs for latent variable estimation. We first discuss the general composite GP structure jointly specified for two GPs along with the full derivative GPs as a special case. Following that, we propose the mathematical conditions for the composite (and derivative) GPs and develop a scalable method through Hilbert space approximations. As our setup, we will start by considering a pair of uni-dimensional output (response) variables $\mathbf{y}_f$ and $\mathbf{y}_g$ and a single (shared) input variable $\mathbf{x}$. Accordingly, we will consider observations (y_{f_i}, y_{g_i}, x_i) for $i = 1, 2, ..., N$ where N is the sample size. We will later extend the notations to consider multi-dimensional outputs and latent variable inputs.

5.3.1 Composite Gaussian processes

Consider scenarios where we want to simultaneously study two different sources of information through two stochastic processes $\mathbf{f}$ and $\mathbf{g}$. We assume that these two processes individually follow a GP such that $f(x) \sim \mathcal{GP}(m_f, k_f)$ and $g(x) \sim \mathcal{GP}(m_g, k_g)$. The GPs are defined by mean functions $m_f = m_f(x)$, $m_g = m_g(x)$ and covariance functions $k_f = k_f(x, x')$ and $k_g = k_g(x, x')$, respectively, over the inputs $x, x' \in \mathbb{R}$. Jointly, we call them composite GPs. We specify them similar to the joint predictive GPs [69] and write the composite GP as

$$\begin{pmatrix} f(x) \\ g(x) \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} m_f \\ m_g \end{pmatrix}, \begin{pmatrix} k_f & k_{fg} \\ k_{gf} & k_g \end{pmatrix} \right), \tag{5.1}$$

where k_{fg} and k_{gf} are functions encoding the relationship (i.e., interactions) between $f(x)$ and $g(x)$, such that, $\text{Cov}(f_i, g_j) = k_{fg}(x_i, x_j)$ and $\text{Cov}(g_i, f_j) = k_{gf}(x_i, x_j)$. In real world scenarios, the functional form of k_{fg} and k_{gf} is often unclear. In such cases, it becomes challenging to specify the full joint covariance structure of a composite GP model. The Hilbert space framework requires knowing the explicit functional forms of k_{fg} and k_{gf}.

The composite GPs have two (univariate) output variables $\mathbf{y}_f$ and $\mathbf{y}_g$. Modeling the common relationship of $\mathbf{x}$ with $\mathbf{y}_f$ and $\mathbf{y}_g$ given the corresponding GPs with independent additive noises can be written as

$$y_{f_i} = f(x_i) + \varepsilon_{f_i} \quad \text{and} \quad y_{g_i} = g(x_i) + \varepsilon_{g_i}, \tag{5.2}$$

where $\varepsilon_{f_i} \sim \mathcal{N}(0, \sigma_f^2)$ and $\varepsilon_{g_i} \sim \mathcal{N}(0, \sigma_g^2)$. Together, this is equivalent to

$$y_{f_i} \mid f \sim \mathcal{N}(f(x_i), \sigma_f^2) \quad \text{and} \quad y_{g_i} \mid g \sim \mathcal{N}(g(x_i), \sigma_g^2). \tag{5.3}$$

Thus, for $\mathbf{y}_f$, when $i \neq j$ we have $\text{Cov}(y_{f_i}, y_{f_j}) = k_f(x_i, x_j)$ and for $i = j$, we have $\text{Cov}(y_{f_i}, y_{f_j}) = \text{Var}(y_{f_i}) = k_f(x_i, x_j) + \sigma_f^2$. This is similar for $\mathbf{y}_g$ involving k_g and σ_g instead.

Under this framework, the functional form of k_f and k_g needs to be chosen. Although we show our theoretical results for a general covariance function, in this paper, we refer to the Matérn family and specifically the Squared exponential (SE) covariance functions [69]. For practical purposes, we prefer SE since it is infinitely differentiable: a property which we use in our derivative GPs discussed below in Section 5.3.2.

5.3.2 Derivative Gaussian processes

Using the framework of composite GPs, joint derivative GPs are a special case where we consider both $\mathbf{f}$ and its derivative $\mathbf{f}^{(1)}$ [60, 77]. The joint derivative GP is used when we wish to model data $\mathbf{y}$ and its derivative $\mathbf{y}^{(1)}$. Concretely, we consider $f(x)$ and its derivative process $f^{(1)}(x)$ such that $f^{(1)}(x) \sim \mathcal{GP}(m_{f^{(1)}}, k^{(1,1)})$ [69, 77], where $k^{(1,1)}(x, x') = \partial_x^1 \partial_{x'}^1 k(x, x')$ denotes the second order partial derivative of $k(x, x')$ differentiated with respect to both x and x'. We model $f(x)$ and $f^{(1)}(x)$ jointly as

$$\begin{pmatrix} f(x) \\ f^{(1)}(x) \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} m_f \\ m_{f^{(1)}} \end{pmatrix}, \begin{pmatrix} k & k^{(1,0)} \\ k^{(0,1)} & k^{(1,1)} \end{pmatrix} \right), \tag{5.4}$$

where $k^{(1,0)} = \partial_x^1 k(x, x')$ and $k^{(0,1)} = \partial_{x'}^1 k(x, x')$. Thus, the off-diagonal covariances are obtained through the Gram matrix $\mathbf{K}^{(1,0)}$ generated by function $k^{(1,0)}$ and its transpose $\mathbf{K}^{(0,1)}$ generated by $k^{(0,1)}$. Based on the aforementioned GP framework, we then model outputs $\mathbf{y}$ and its derivative $\mathbf{y}^{(1)}$ which can be obtained by replacing $\mathbf{y}_f$ and $\mathbf{y}_g$ in Eq.(5.3) by $\mathbf{y}$ and $\mathbf{y}^{(1)}$ respectively. The independent additive noises with error SDs σ_f and σ_g then correspond to $\mathbf{y}$ and $\mathbf{y}^{(1)}$ respectively.

Although we only show the second order partial derivatives in Eq.(5.4), this structure holds for higher order derivative functions as well. Considering the general derivative structure of covariance functions k and their corresponding spectral densities S_k [31] (see Section 5.3.4 for details) we propose the following result:

Proposition 1. *Let $k(x, x') = k(r)$, where $r = x - x'$, be a stationary covariance kernel on $\mathbb{R}$ with spectral density $S_k \in L^1(\mathbb{R})$ such that, for some integer $m \geq 1$,*

$$\int_{\mathbb{R}} \|\omega\|^m S_k(\omega)\, d\omega < \infty.$$

For $a, b \in \mathbb{N}_0$ and $m \geq 1$ such that $a + b \leq m$, define

$$k^{(a,b)}(r) = \partial_x^a \partial_{x'}^b k(x, x').$$

Then

$$k^{(a,b)}(r) = (-1)^b \partial_r^{a+b} k(r).$$

The conditions for this general derivative function $k^{(a,b)}(x, x')$ to be a covariance functions is thus follows:

Proposition 2 (Conditions for $k^{(a,b)}$ to be a covariance kernel)**.** *Assume the kernel k is stationary and isotropic with spectral density $S_k(\omega)$ and that $a + b \leq m$. Then the derivative function $k_{a,b}$ is a positive semidefinite covariance kernel if and only if the following set of conditions hold:*

(1) $a + b$ is even.

(2) a, b are such that $a + 3b \equiv 0 \pmod 4$.

In particular, $a = b$ always satisfies these conditions, so $k^{(a,a)}$ is a covariance kernel.

While we show the case of uni-dimensional input x, our propositions hold for p-dimensional inputs $\mathbf{x} \in \mathbb{R}^p$. We present the proofs in Appendix C.1.

For a SE covariance function the derivative covariance structure is thus obtained as

$$\begin{aligned} k(x_i, x_j) &= \alpha^2 \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right), \\ k^{(1,0)}(x_i, x_j) &= \alpha^2 \frac{(x_i - x_j)}{\rho^2} \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right), \\ k^{(1,1)}(x_i, x_j) &= \frac{\alpha^2}{\rho^4}(\rho^2 - (x_i - x_j)^2) \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right). \end{aligned} \tag{5.5}$$

where $\rho > 0$ and $\alpha > 0$ are the length-scale and GP marginal SD respectively. Further details on the derivative covariance structure and other derivative Matérn covariance functions can be found in [60].

5.3.3 Partial composite Gaussian processes

In real world applications, the exact relationship between $\mathbf{f}$ and $\mathbf{g}$ often remains unknown. As previously mentioned, under the composite GP framework, it becomes challenging to define the functional forms of k_{fg} and k_{gf} (see Section 5.3.1) thus affecting the overall choice of the covariance function structure. In cases like the derivative GPs where the forms of k_{fg} and k_{gf} are known through $k^{(1,0)}$ and $k^{(0,1)}$ respectively, we face limitations in approximating the model. Under the Hilbert space framework, we would require all four quadrants of the joint covariance matrix to satisfy the conditions of Proposition 2 for a block-wise approximation strategy. While the diagonal functions indeed satisfy these conditions, the functions in the off-diagonals do not (see Appendix C.1).

In such cases, for practical reasons, we assume $\mathbf{f}$ and $\mathbf{g}$ are independent, thus relaxing the overall composite covariance structure. We refer to the composite GPs with such a covariance structure

as partial composite GPs or pcGPs. The pcGP is therefore modified from Eq.(5.1) as

$$\begin{pmatrix} f(x) \\ g(x) \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} m_f \\ m_g \end{pmatrix}, \begin{pmatrix} k_f & 0 \\ 0 & k_g \end{pmatrix} \right), \tag{5.6}$$

where the off-diagonals in the joint covariance function indicates independence between $\mathbf{f}$ and $\mathbf{g}$. The rest of the model remains unaltered. We will focus only on this pcGP structure in this paper and the above notations will be extended to multi-output and latent variable GPs in Section 5.3.6. The properties of the two GPs depend on the choice of the corresponding covariance functions. As a specific case of pcGPs, we thus introduce partial derivative GPs (pdGPs) that are similarly modified from Eq.(5.4). Concretely, we arrive at the pdGP formulation by considering $\mathbf{f}$ along with the covariance function k followed by substituting $g(x) = f^{(1)}(x)$ and subsequently $k_g = k^{(1,1)}$ in Eq.(5.6).

5.3.4 Partial composite Hilbert space Gaussian processes

Exact GPs are known to have a computational complexity of $O(N^3)$ where N is the sample size, thus prohibiting applications on cases where N is large. We overcome this scalability issue through HSGP methods [61, 72, 78], where we approximate the covariance function with its spectral density. Briefly, following Bochner's theorem [31] and Wiener-Khintchine theorem [23], any stationary covariance function $k(r)$ with $r = x - x'$ can be written as

$$S_k(\omega) = \int_{\mathbb{R}} \exp(-i\omega r) k(r) dr \tag{5.7}$$

where $\omega \in \mathbb{R}$ are the inputs in the frequency domain.

Thus, the spectral density for a SE covariance function [31] is given as

$$S_{k_f}(\omega) = \sqrt{2\pi}\alpha_f^2 \rho_f \exp\left(-\frac{1}{2}\rho_f^2 \omega^2\right) \tag{5.8}$$

where ρ_f and α_f are the covariance function hyperparameters for a GP f.

The HSGP procedure starts by defining a closed set $\Omega \subset [-L, L] \subset \mathbb{R}$ that contains the vector of inputs $\mathbf{x}$. We then write any stationary covariance function with $x, x' \in \Omega$ as

$$k(x, x') = \sum_{j=1}^{\infty} S(\sqrt{\lambda_j}) \phi_j(x) \phi_j(x'), \tag{5.9}$$

where S is the spectral density of the covariance function k. Following the Dirichlet conditions [78], the sets of eigenvalues $\{\lambda_j\}_{j=1}^{\infty}$ and eigenfunctions $\{\phi_j\}_{j=1}^{\infty}$ are given by

$$\begin{aligned} \lambda_j &= \left(\frac{j\pi}{2L}\right)^2, \\ \phi_j(x) &= \sqrt{\frac{1}{L}} \sin\left(\sqrt{\lambda_j}(x + L)\right). \end{aligned} \tag{5.10}$$

The eigenvalues and eigenfunctions are independent of the choice of covariance functions, and

any effects of the hyperparameters for a chosen covariance function are informed through the corresponding spectral density. We approximate the covariance function using a linear combination of its basis functions along with the spectral density, eigenvalues and eigenfunctions. Considering the first M number of basis terms, we obtain from Eq.(5.9)

$$k(x, x') \approx \sum_{j=1}^{M} S(\sqrt{\lambda_j})\phi_j(x)\phi_j(x'). \tag{5.11}$$

In case of the pcGPs with $\mathbf{f}$ and $\mathbf{g}$ having covariance functions k_f and k_g, we first derive their corresponding spectral densities S_{k_f} and S_{k_g}. We then apply the Hilbert space approximation such that

$$\begin{pmatrix} f(x) \\ g(x) \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} m_f \\ m_g \end{pmatrix}, \begin{pmatrix} \tilde{k}_f & 0 \\ 0 & \tilde{k}_g \end{pmatrix} \right) \tag{5.12}$$

where $\tilde{k}_f = \sum_{j=1}^{M} S_{k_f}(\sqrt{\lambda_j})\phi_j(x)\phi_j(x')$ and $\tilde{k}_g = \sum_{j=1}^{M} S_{k_g}(\sqrt{\lambda_j})\phi_j(x)\phi_j(x')$ are the Hilbert space approximated covariance functions corresponding to k_f and k_g respectively. Note that the input space is shared between f and g and thus the eigenvalues λ and eigenfunctions ϕ are evaluated on the same set of inputs x. So, any difference in the covariance functions between f and g (different hyperparameters in our case) are, again, only informed through their corresponding spectral densities.

The full covariance matrices $\mathbf{K}_f$ and $\mathbf{K}_g$ that are generated from k_f and k_g respectively for inputs x_i, $i \in \{1, \ldots, N\}$ are approximated by its finite basis function as

$$\mathbf{K}_f \approx \mathbf{\Phi}\mathbf{\Delta}_f\mathbf{\Phi}^T \quad \text{and} \quad \mathbf{K}_g \approx \mathbf{\Phi}\mathbf{\Delta}_g\mathbf{\Phi}^T, \tag{5.13}$$

where $\mathbf{\Delta}_f = \text{diag}(S_{k_f}(\sqrt{\lambda_1}), \ldots, S_{k_f}(\sqrt{\lambda_N}))$, $\mathbf{\Delta}_g = \text{diag}(S_{k_g}(\sqrt{\lambda_1}), \ldots, S_{k_g}(\sqrt{\lambda_N}))$ and $\mathbf{\Phi} \in \mathbb{R}^{N \times M}$ is the matrix of eigenfunctions

$$\mathbf{\Phi} = \mathbf{\Phi}(\mathbf{x}) = \begin{bmatrix} \phi_1(x_1) & \ldots & \phi_M(x_1) \\ \vdots & \ddots & \vdots \\ \phi_1(x_N) & \ldots & \phi_M(x_N), \end{bmatrix} \tag{5.14}$$

for a vector of input points $\mathbf{x} = (x_1, \ldots, x_N)$. We then replace the covariance matrix $\mathbf{K}_f$ and $\mathbf{K}_g$ by its basis approximated eigenvalue decompositions and specify the GPs $\mathbf{f}$ and $\mathbf{g}$ following the linear representation from [61, 72] as

$$\begin{aligned} f(\mathbf{x}) &\approx \mu_f + \sum_{j=1}^{M} \left(S_{k_f}(\sqrt{\lambda_j}) \right)^{1/2} \mathbf{\Phi}(\mathbf{x})\beta_{fj}, \\ g(\mathbf{x}) &\approx \mu_g + \sum_{j=1}^{M} \left(S_{k_g}(\sqrt{\lambda_j}) \right)^{1/2} \mathbf{\Phi}(\mathbf{x})\beta_{gj}, \end{aligned} \tag{5.15}$$

where $\beta_{fj} \sim \mathcal{N}(0, 1)$ and $\beta_{gj} \sim \mathcal{N}(0, 1)$ for $j \in \{1, \ldots, M\}$. This results pcHSGPs with $\mathbf{f}$ and $\mathbf{g}$ having eigen-decomposed approximate covariance functions. Using pcHSGP, we then simultaneously model outputs $\mathbf{y}_f$ and $\mathbf{y}_g$ with a shared input $\mathbf{x}$ as shown in Eq.(5.3).

5.3.5 Partial derivative Hilbert space Gaussian processes

Under the Hilbert space methods, we approximate the covariance functions of the GP $\mathbf{f}$ and its derivative process $\mathbf{f}^{(1)}$ with its spectral density. Based on the Proposition 1, related to the general structure of derivative covariance functions (Section 5.3.2), we state the following:
Proposition 3 (Spectral representation of derivative kernels)**.** *Let $k^{(a,b)}(x, x') = k^{(a,b)}(r)$ be a stationary covariance kernel, then its spectral density $S^{(a,b)}(\omega)$ is given by*

$$S^{(a,b)}(\omega) = (i\omega)^a(-i\omega)^b S_k(\omega).$$

See Appendix C.1 for the proof. Additionally, we derive the spectral densities and the conditions under which they satisfy Proposition 3 for SE as well as the general Matérn family of covariance functions (see Corollary 1 and 2 in Appendix C.1). Furthermore, our choice of k preserves the assumptions of $k^{(1,1)}$ being isotropic and subsequently $S_{k^{(1,1)}}$ having an analytic functional form, thus satisfying the Hilbert space approximation conditions [78].

Taking the example of the derivative SE covariance function $k^{(1,1)}$ shown in Section 5.3.2, we thus obtain the spectral density of the using Proposition 3:

$$S_{k^{(1,1)}}(\omega) = \omega^2 S_k(\omega) = \sqrt{2\pi}\omega^2\alpha^2\rho\exp\left(-\frac{1}{2}\rho^2\omega^2\right), \tag{5.16}$$

where $S_k(\omega)$ is the spectral density of the standard SE covariance function (as seen in Eq.(5.8)).

We resort again to the pcGP structure for reasons described in Section 5.3.3. Thus, the derivative HSGP is now specified as

$$\begin{pmatrix} f(x) \\ g(x) \end{pmatrix} \sim \mathcal{GP}\left(\begin{pmatrix} m_f \\ m_g \end{pmatrix}, \begin{pmatrix} \tilde{k} & 0 \\ 0 & \tilde{k}^{(1,1)} \end{pmatrix}\right) \tag{5.17}$$

such that $\tilde{k} = \sum_{j=1}^{M} S_k(\sqrt{\lambda_j})\phi_j(x)\phi_j(x')$ and $\tilde{k}^{(1,1)} = \sum_{j=1}^{M} S_{k^{(1,1)}}(\sqrt{\lambda_j})\phi_j(x)\phi_j(x')$ are the Hilbert space approximated covariance functions corresponding to k and $k^{(1,1)}$ respectively. The rest of the steps for the approximation procedure remains the same. Using pdHSGPs, we overcome the practical limitations of derivative GPs outlined in [60]. However, in pdHSGPs, we don't model and learn the functional relationship between $\mathbf{f}$ and $\mathbf{f}^{(1)}$ through the first order partial derivatives of the covariance functions. Thus, pdHSGPs are, to some degree, misspecified in cases where the underlying data generating process is a full derivative GP model. We show the degree of trade-off for this misspecification as a means to a scalable solution for derivative GPs in Section 5.4.

5.3.6 Extending the partial composite structure

We extend the pcHSGPs and pdHSGPs to model multi-dimensional outputs and latent variable inputs. In this paper, we follow a similar framework pertaining to the multi-output latent variable models considered in [60, 61].

Multi-dimensional outputs

We specify the multi-output version of the composite HSGPs with response variables $(\mathbf{y}_{f_1}, \ldots, \mathbf{y}_{f_D})$ and $(\mathbf{y}_{g_1}, \ldots, \mathbf{y}_{g_D})$ over $D > 1$ output dimensions [69], using $y_{f_{di}}$ and $y_{g_{di}}$ to denote the response for d^{th} dimension and i^{th} sample. As the usual approach, we first set up D independent, univariate Gaussian processes $\mathbf{f}_d$ and $\mathbf{g}_d$ each with their own set of hyperparameters $\boldsymbol{\theta}_{f_d}$ and $\boldsymbol{\theta}_{g_d}$ [60, 61, 85]. We extend the composite HSGPs to multi-output GPs by modifying Eq.(5.15) such that

$$\begin{aligned} f_d(\mathbf{x}) &\approx \mu_{f_d} \sum_{j=1}^{M} \left(S_{k_{f_d}}(\sqrt{\lambda_j})\right)^{1/2} \boldsymbol{\Phi}(\mathbf{x})\beta_{f_{jd}}, \\ g_d(\mathbf{x}) &\approx \mu_{g_d} + \sum_{j=1}^{M} \left(S_{k_{g_d}}(\sqrt{\lambda_j})\right)^{1/2} \boldsymbol{\Phi}(\mathbf{x})\beta_{g_{jd}}, \end{aligned} \tag{5.18}$$

where $\beta_{f_{jd}} \sim \mathcal{N}(0,1)$ and $\beta_{g_{jd}} \sim \mathcal{N}(0,1)$ for the corresponding GP approximations. The dimensions of $\boldsymbol{\Phi}$ thus remains same, since it depends only on sample size N and number of basis functions M. With the multi-dimensional outputs D, the exact GPs have a computational complexity of $O(N^3D + ND^2)$ where N is the sample size of outputs. Through this HSGP formulation using M number of basis functions, we sharply decrease it to $O(NMD + ND^2)$.

The univariate GPs are then related to one another by linearly combining them with a (D-dimensional) across-dimension uniform correlation matrix $\mathbf{C}_f$ and $\mathbf{C}_g$ [12, 85]. Specifically, for each i^{th} sample, we obtain a vector of across-dimension correlated GP values as

$$\begin{pmatrix} f_1^*(x_i) \\ \ldots \\ f_D^*(x_i) \end{pmatrix} = \mathbf{A}_f \times \begin{pmatrix} f_1(x_i) \\ \ldots \\ f_D(x_i) \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} g_1^*(x_i) \\ \ldots \\ g_D^*(x_i) \end{pmatrix} = \mathbf{A}_g \times \begin{pmatrix} g_1(x_i) \\ \ldots \\ g_D(x_i) \end{pmatrix}, \tag{5.19}$$

where $\mathbf{A}_f$ and $\mathbf{A}_g$ are the Cholesky factors of $\mathbf{C}_f$ and $\mathbf{C}_g$ respectively such that $\mathbf{C}_f = \mathbf{A}_f\mathbf{A}_f^T$ and $\mathbf{C}_g = \mathbf{A}_g\mathbf{A}_g^T$. This way, multi-output GPs combine two dependency structures, one within dimensions (and across observations) as expressed by the univariate GPs through corresponding covariance functions and one across output dimensions (but within observations) as expressed by $\mathbf{C}_f$ and $\mathbf{C}_g$ for both the GPs. Adding independent Gaussian noise to our derivative multi-output GP model, we extend Eq.(5.3) for all d and i:

$$y_{f_{di}} \mid f_d^* \sim \mathcal{N}(f_d^*(x_i), \sigma_{f_d}^2) \quad \text{and} \quad y_{g_{di}} \mid g_d^* \sim \mathcal{N}(g_d^*(x_i), \sigma_{g_d}^2). \tag{5.20}$$

The structure remains same for the derivative version of the composite HSGP where we again consider $g(x) = f^{(1)}(x)$ along with the corresponding changes in the covariance function (and spectral densities).

Latent variable inputs

Within latent-variable composite HSGPs, a single input vector $\mathbf{x}$ shared between $\mathbf{f}$ and $\mathbf{g}$ (and $\mathbf{f}^{(1)}$ in case of derivative HSGPs) is considered as unobserved and treated similar to other estimable parameters. To achieve this, we first consider an observed quantity $\tilde{\mathbf{x}}$ that acts as a noisy measurement to the latent inputs $\mathbf{x}$. From a Bayesian perspective, $\tilde{\mathbf{x}}$ is a quantity based

on which we define a prior for the latent $\mathbf{x}$, which is then refined by the HSGPs learning from $\mathbf{y}_f$ and $\mathbf{y}_g$ (and $\mathbf{y}^{(1)}$ for derivative HSGPs). Specifically, we assume that the measurements $\tilde{\mathbf{x}}$ are Gaussian with known measurement SD s such that we can write for each observation i:

$$\tilde{x}_i \sim \mathcal{N}(x_i, s^2). \tag{5.21}$$

The vector of latent inputs $\mathbf{x}$ is then passed on to the approximation step in Eq.(5.15) (and subsequently Eq.(5.18) for the multi-output case). We call the resulting model (multi-output) latent variable HSGPs. Latent variable GPs, exact or approximate, are more difficult to fit than their manifest counterparts. The primary reasons are the substantial increase in the number of estimable parameters as well as identification issues arising due to both $\mathbf{x}$ and $\boldsymbol{\rho}$ now being treated as unknown. More details on these issues are discussed in [60, 61]. Briefly, we overcome these issues by pooling information through multiple output dimensions for a single latent $\mathbf{x}$. The noisy measurements $\tilde{\mathbf{x}}$ additionally help identifying the length-scale $\boldsymbol{\rho}$ and latent $\mathbf{x}$.

5.4 Simulation study

For latent variable models, we lack ground truth values in real-world scenarios and thus it is challenging to validate such models based on real data alone. We therefore design simulation studies where we validate our developed methods against simulated ground truths. We compare pcHSGPs and pdHSGPs to their exact counterparts on their latent variable estimation accuracy and inference speed. The overall simulation study design is inspired from [60, 61], but extended to include the pcHSGPs, pdHSGPs, and their exact models pcGP and pdGP respectively. We fit all the models involved using full Bayesian inference via MCMC sampling in Stan [80]. The details of the model inference procedure are presented in Appendix C.2.

5.4.1 Data generating process

We consider two data generating processes as our simulation scenarios. In the first scenario, we generate the data from an exact pcGP (from Eq.(5.6)). Under this scenario, we showcase the advantages of using the approximate pcHSGP model when the true underlying data is the exact pcGP. In the second scenario, we generate data using a full derivative GP (dGP) (see Eq.(5.4)). For the dGP data scenario, we compare the levels of model misspecification based on two counts: partial covariance structure (via pdGPs and pdHSGPs) and covariance function mismatch (via pcGPs and pcHSGPs).

In the above scenarios, we have a sample size of $N = 20$ since exact GPs are considered among the set of models to be compared. Thus, as third scenario, we consider dGP data with higher $N = 100$ where we only compare different HSGPs. Under this scenario, in addition to pcHSGP and pdHSGPs, we consider single HSGPs (sHSGP) and single derivative HSGP (sdHSGP) where we only have single GP functions $\mathbf{f}$ and $\mathbf{f}^{(1)}$ respectively. Including the sHSGP and sdHSGP answers the question regarding the advantages of composite data sources versus a simpler single source of information. Although HSGPs could easily handle much higher N as we show in

our case study (in Section 5.5), we use only use $N = 100$ to reduce the overall runtime of the simulations. An outline of our simulation scenarios are presented in Table 5.1.

Table 5.1: Simulation study design

Simulation scenario	Sample size (N)	Data generating process	Covariance function
1	20	pcGP	partial composite SE
2	20	dGP	joint derivative SE
3	100	dGP	joint derivative SE

Note: All simulation scenarios consists of output dimensions $D = 5, 10$ and 20. The data generating processes are partial composite GP (pcGP) and joint derivative GP (dGP). The covariance functions are based on Squared Exponential (SE) function.

Our data generating conditions are specified in a way that ensures a fair amount of non-linearity (via length-scales) as well as a good signal-to-noise ratio (via marginal and error SDs) in the simulated data. Concretely, for the pcGP data generating process, we sample multi-dimensional length-scales $\boldsymbol{\rho}_f \sim \text{Normal}^+(1, 0.05^2)$ and $\boldsymbol{\rho}_g \sim \text{Normal}^+(0.7, 0.05^2)$, GP marginal SDs $\boldsymbol{\alpha}_f \sim \text{Normal}^+(3, 0.25^2)$ and $\boldsymbol{\alpha}_g \sim \text{Normal}^+(2, 0.25^2)$, and error SDs $\boldsymbol{\sigma}_f \sim \text{Normal}^+(1, 0.25^2)$ and $\boldsymbol{\sigma}_g \sim \text{Normal}^+(0.75, 0.25^2)$. In case of the dGP data, we first induce a scale difference between the outputs $\mathbf{y}$ and derivatives $\mathbf{y}^{(1)}$ (as well as $\mathbf{f}$ and $\mathbf{f}^{(1)}$ subsequently) using a scaling proportion $\lambda = 10$. Thus, we re-purpose the GP marginal SDs and error SDs of composite GPs to be $\boldsymbol{\alpha}_f = \boldsymbol{\alpha}\,, \boldsymbol{\alpha}_g = \boldsymbol{\alpha}^{(1)}$ and $\boldsymbol{\sigma}_f = \boldsymbol{\sigma}\,, \boldsymbol{\sigma}_g = \boldsymbol{\sigma}^{(1)}$ to remain coherent with the derivative GP notations. We induce this scale difference by first sampling marginal and error SDs for $\mathbf{f}^{(1)}$ and $\mathbf{y}^{(1)}$ where $\boldsymbol{\alpha}^{(1)} \sim \text{Normal}^+(3, 0.25^2)$ and $\boldsymbol{\sigma}^{(1)} \sim \text{Normal}^+(1, 0.25^2)$. Then for the outputs $\mathbf{y}$ (and functions $\mathbf{f}$), we set $\boldsymbol{\alpha} = \lambda\boldsymbol{\alpha}^{(1)}$ and $\boldsymbol{\sigma} = \lambda\boldsymbol{\sigma}^{(1)}$. Under this data generating process, we only have a single length-scale $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.05^2)$. More details regarding the implications and relevance of this scaling proportion λ can be found in [60].

In pcGP data scenario, we sample across-output dimension correlation matrices $\mathbf{C}_f$ and $\mathbf{C}_g$ from the LKJ($\eta = 1$) distribution [52]. A value of $\eta = 1$ implies $\mathbf{C}_f$ and $\mathbf{C}_g$ to be uniformly distributed within the set of all correlation matrices of dimension D. For the dGP data scenario, we only have a single large between-dimension (output) correlation matrix $\mathbf{C} \sim \text{LKJ}(\eta = 1)$. For both data generating processes, we sample the constant mean functions m_f and m_g (which is $m_{f^{(1)}}$ for dGP data) from $\text{Normal}^+(0, 5^2)$. We generate the ground truth latent inputs for the simulation scenarios as $x_i \sim \text{Uniform}(0, 10)$, such that $i = 1, \ldots, N$. Furthermore, we assume a prior measurement SD of the noisy $\tilde{\mathbf{x}}$ as $s = 0.3$ (see Section 5.3.6). For the number of output dimensions D, we consider the cases $D = 5, 10$ and 20. We perform 50 simulation trials for each of the choices of D for every simulation scenario.

5.4.2 Model specifications

We fit a set of exact GPs and their approximated HSGPs for the simulation scenarios discussed above. Under the pcGP data scenario, we only compare the true model against its approximation pcHSGP. In this case, we check the benefits of using Hilbert space approximations for composite GP models. For the dGP data scenario, we compare pcGP and pdGP along with

their approximations pcHSGP and pdHSGP respectively. In the third scenario with dGP data and $N = 100$ we only involve HSGPs. In this case, we include sHSGP and sdHSGP having a single source of information in addition to pcHSGPs and pdHSGPs. Including sHSGPs and sdHSGPs checks the advantages of composite models with two sources of data as opposed to simpler HSGPs with a single data source. The entire set of comparative models along with their features are presented in Table 5.2.

Table 5.2: GP model specifications involved in our simulation study

Model name	Model type	Simulation scenarios	Composite	Derivative	Partial
pcGP	E	1, 2	✓	✗	✓
dGP	E	2	✓	✓	✗
pdGP	E	2 k	✓	✓	✓
pcHSGP	A	1, 2, 3	✓	✗	✓
pdHSGP	A	2, 3	✓	✓	✓
sHSGP	A	3 k	✗	✗	✗
sdHSGP	A	3	✗	✓	✗

Note: Model names denote partial composite GPs (pcGP) and partial derivative GPs (pdGP) and their approximations pcHSGP and pdHSGP respectively. Additionally, we have the joint derivative GP (dGP), single HSGP (sHSGP) and single derivative HSGP (sdHSGP). The models are either exact (E) or approximate (A) in nature. We use the simulation scenario numbering from Table 5.1.

The priors for all model parameters are the same as those used in the data generating process, a requirement for the uncertainty calibration tests (shown in Appendix C.5). Thus, in the pcGP data simulation scenario, the models pcGP and pcHSGP length scales have multi-dimensional hyperparameters $\boldsymbol{\rho}_f \sim \text{Normal}^+(1, 0.05^2)$ and $\boldsymbol{\rho}_g \sim \text{Normal}^+(0.7, 0.05^2)$, GP marginal SDs $\boldsymbol{\alpha}_f \sim \text{Normal}^+(3, 0.25^2)$ and $\boldsymbol{\alpha}_g \sim \text{Normal}^+(2, 0.25^2)$, and error SDs $\boldsymbol{\sigma}_f \sim \text{Normal}^+(1, 0.25^2)$ and $\boldsymbol{\sigma}_g \sim \text{Normal}^+(0.75, 0.25^2)$. Under the dGP data scenario, the pdGP and pdHSGP involving derivative SE structure only have a single length scale $\boldsymbol{\rho} \sim \text{Normal}^+(1, 0.05^2)$. The marginal SDs $\boldsymbol{\alpha} \sim \text{Normal}^+(30, 2.5^2)$ and $\boldsymbol{\alpha}^{(1)} \sim \text{Normal}^+(3, 0.25^2)$. The error SDs are specified to follow $\boldsymbol{\sigma} \sim \text{Normal}^+(10, 2.5^2)$ and $\boldsymbol{\sigma}^{(1)} \sim \text{Normal}^+(1, 0.25^2)$. Under this scenario, in case of pcGP and pcHSGP, the priors for both length-scales $\boldsymbol{\rho}_f$ and $\boldsymbol{\rho}_g$ remain the same as $\boldsymbol{\rho}$. The marginal $\boldsymbol{\alpha}_f$, $\boldsymbol{\alpha}_g$ and error SDs $\boldsymbol{\sigma}_f$, $\boldsymbol{\sigma}_g$ follow the same prior distributions as $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^{(1)}$ and $\boldsymbol{\sigma}$, $\boldsymbol{\sigma}^{(1)}$ respectively. For the dGP data scenario with $N = 100$, we have the sHSGP and sdHSGP, having only a single covariance structure, involves a single set of hyperparameters $\boldsymbol{\rho}$, $\boldsymbol{\alpha}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\rho}$, $\boldsymbol{\alpha}^{(1)}$, $\boldsymbol{\sigma}^{(1)}$ respectively. The respective priors for these hyperparameters remain the same as described above.

Based on the suggestions in [61, 72], we select the boundary conditions L for all the HSGP models involved, by multiplying a scaler adjustment $c = 1.25$ to the range of the input prior $\tilde{\mathbf{x}}$. This way, we ensure that we have a closed interval $[-L, L]$ that prevents input values $\mathbf{x}$ to be near the boundaries. While there is an empirical relation that determines the minimum number of required basis functions M [72], it is only applicable for standard Matérn class of covariance functions. Since it has not been extended for composite (and derivative) covariance functions, we select $M = 30$ for the all the simulation scenarios based on the results of [61], which is enough to capture the non-linearity for both SE and derivative SE functions through

their spectral approximations.

We implemented all models in Stan [80] and conducted the simulation studies with the `rstan` interface [79]. The models are fitted with a single MCMC chain of 2000 iterations of which the first 1000 are discarded as warm-up. [60] show that model convergence is similar for multiple chains when fitting latent variable GPs for the SE and derivative SE covariance functions. Thus, we run only a single chain per model to parallelize over the 50 trials to reduce overall computation times.

5.4.3 Latent variable estimates

We evaluate the latent variable estimation accuracy for our proposed models under all simulation scenarios. To this end, we compare posterior samples of latent $\mathbf{x}$ denoted by $\mathbf{x}_{post}$ from our fitted models to their true values $\mathbf{x}_{true}$ using $\text{RMSE}(\mathbf{x}_{post}) = \sqrt{\mathbb{E}\left((\mathbf{x}_{post} - \mathbf{x}_{true})^2\right)}$. Under the different simulation conditions described before, we study the effects of our various model choices on $\text{RMSE}(\mathbf{x}_{post})$. We summarize these effects through a multilevel model setup described in Appendix C.4. Our results show the HSGPs out-perform their exact GP counterparts in terms of RMSE, for latent variable estimation, in all simulation scenarios. Specifically, for the pcGP data

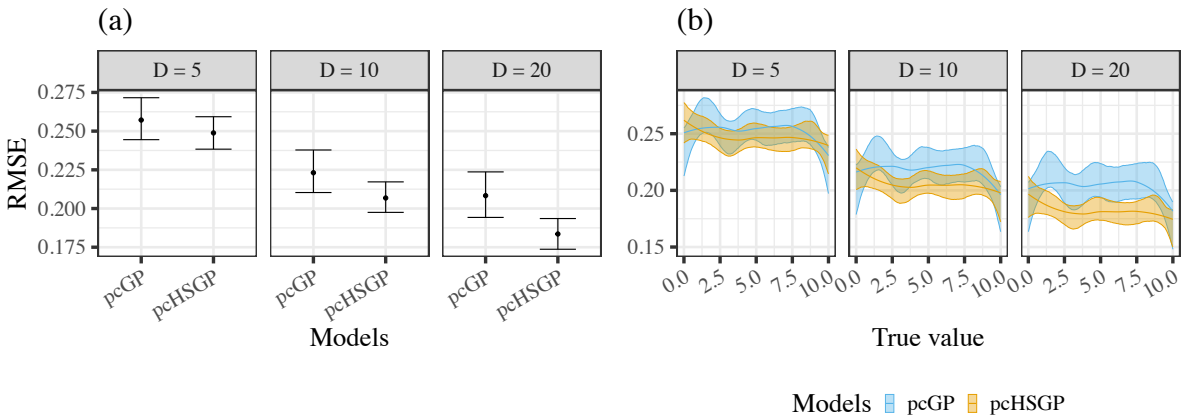

Figure 5.1: *pcGP data scenario: a) RMSE on recovery of latent inputs for the data generating process pcGP and its approximation pcHSGP. b) Model performance in terms of RMSE for recovering various true values across the input space. Figures are shown for output dimensions* $D = 5, 10$ *and* 20.

scenario (Fig. 5.1), we see up to 14% decrease in RMSE (in case of higher output dimensions) with pcHSGPs when compared to the exact pcGP (true data generating process) while also drastically speeding up model fitting.

In Fig. 5.2, we demonstrate the effects of estimating latent variables under several covariance function misspecifications. When compared to the ground truth dGP, naturally, we see an increase in overall RMSE for estimating latent $\mathbf{x}$ for both pcGP and pdGP. This is a direct consequence of using a partial covariance structure to model data generated from a joint derivative structure, not to mention a different covariance function with the pcGP. However, the respective HSGPs for the partial GPs readily overcomes this by showing a greater accuracy (as compared to their exact counterparts) in estimating latent variable inputs. Concretely, the pcHSGPs show an overall decrease in RMSE as compared to the exact pcGP by up to 10%. As for the

pdHSGP, we see an overall decrease of up to 33% when compared to the exact pdGP model. As for estimation speed, for a dataset with $N = 20$ and $D = 20$, the full derivative GP took on average 6.05 hrs. The exact pcGPs and pdGPs took 2.03 hrs and 3 hrs respectively. Both the pcHSGPs and pdHSGPs on the other hand only took 0.22 hrs to fit the same dataset Under this dGP data scenario with $N = 20$, while the pcHSGPs and pdHSGPs do not yet reach the same accuracy as the ground truth model dGP, they have a clear advantage in terms of model fitting speed.

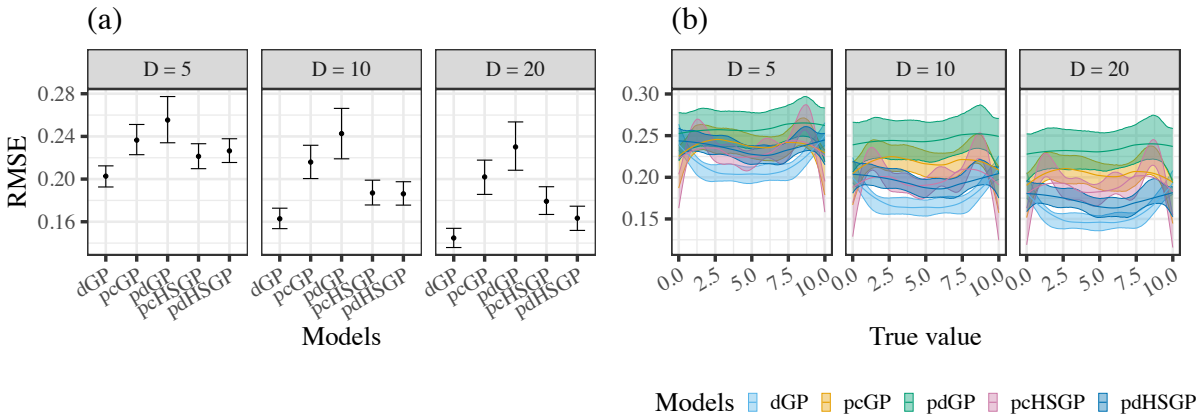

Figure 5.2: *dGP data scenario: a) RMSE on recovery of latent inputs for all fitted models. Models pcGP and pcHSGP refer to partial composite GPs (exact and approximate). pdGP and pdHSGP refers to the exact and approximate partial derivative GPs. b) Model performance in terms of RMSE for recovering various true values across the input space. Figures are shown for output dimensions* $D = 5, 10$ *and* 20.

The results from the dGP data with $N = 100$ scenario (in Fig. 5.3) demonstrates the effects on various choices of HSGPs on latent variable estimation under a larger sample size. In this simulation scenario, we additionally considered models with a single covariance function sHSGP and sdHSGP analyzing either the outputs $\mathbf{y}$ or their derivatives $\mathbf{y}^{(1)}$ separately. We readily see that the composite models pcHSGP and pdHSGP outperforms sHSGP and sdHSGPs (having up to 42% lower RMSE) in estimating latent $\mathbf{x}$.

Through the results of this simulation scenario, we make a couple of important observations. Firstly, among the sHSGP and sdHSGP, the latter outperforms the former in estimating latent $\mathbf{x}$. This is due to the standard SE covariance function being inadequate in modeling the data generated from a much more complex dGP structure. Secondly, both pcHSGP and pdHSGP perform almost equally well, under the same data generating condition but with the higher $N = 100$ sample size. Further, upon comparing the results across Fig. 5.2 and Fig. 5.3, we see that pcHSGPs and pdHSGPs obtain a much lower RMSE of up to 43% (under higher sample size) as compared to ground truth dGP that can only be a feasible modeling option under low sample sizes. For $N = 100$ (and $D = 20$), the pcHSGPs and pdHSGPs took 1.26 hrs and 1.15 hrs, respectively. Further results including model convergence, uncertainty calibration for latent variable estimates, as well as hyperparameter estimation accuracy are provided in Appendix C.3, C.5 and C.6 respectively.

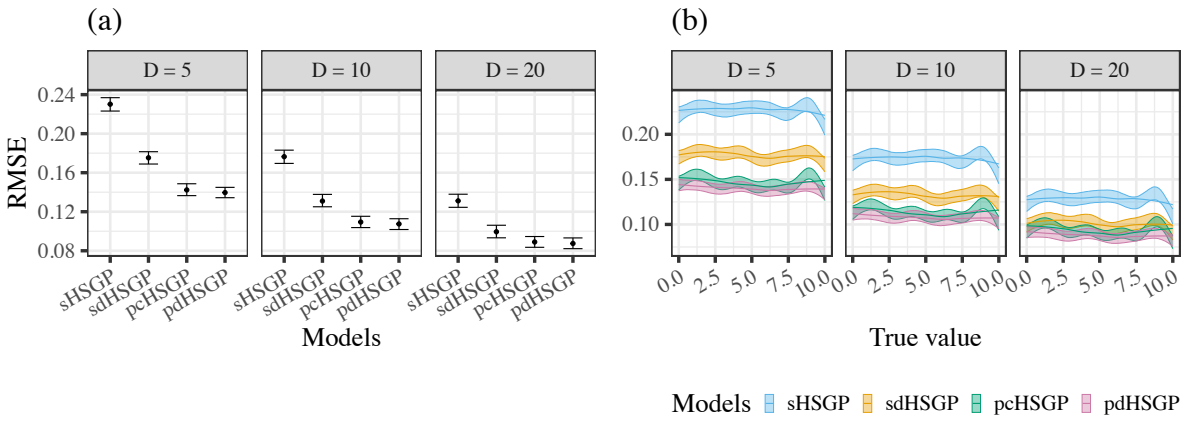

Figure 5.3: *dGP scenario: a) RMSE on recovery of latent inputs for different Hilbert space GPs. sHSGP and sdHSGPs depict models with either SE or derivative SE covariance function based on a single data source. They are compared the partial composite pcHSGPs and derivative pdHSGP models. b) Model performance in terms of RMSE for recovering various true values across the input space. Figures are shown for output dimensions* $D = 5, 10$ *and* 20.

5.5 Real-world case study

We demonstrate our proposed methods on a real-world single-cell RNA sequencing dataset involving the maturation process of erythroid cells from progenitor cells [66]. Single-cell RNA sequencing is a technique that allows the measurement of RNA molecule abundance in single cells [39]. Through these measurements, we study the description of the underlying cells on a molecular level. By arranging all cells along a latent trajectory, it is possible to uncover and characterize an entire biological process. In our case study, the biological process contains two blood progenitors and three maturing erythroid cell stages. The observed process describes a trajectory starting from blood progenitors which develop into matured erythroid cells without any bifurcations or cyclical dynamics. The dataset contains several experimental time points being the time points of sample collection from original embryonic tissue, which correlate with the maturation process of the cells.

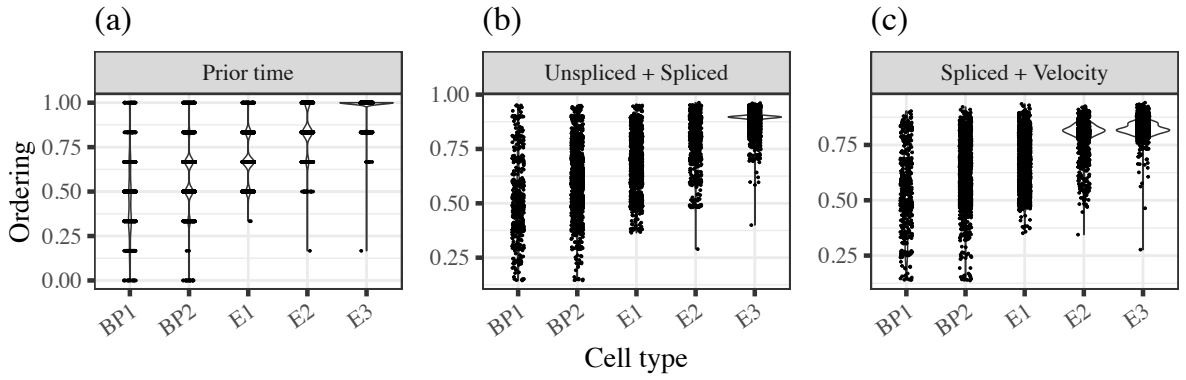

Figure 5.4: *a) Distribution of discrete experimental time by cell type. b) Posterior latent continuous cell ordering from modeling unspliced and spliced gene expression using pcHSGP. c) Posterior latent continuous cell ordering from modeling spliced gene expression and RNA velocity using pdHSGP. The x-axis denotes the cell types blood progenitors (BP) and erythroid (E).*

The pre-annotated dataset was acquired from the *scvelo* [9] package and contains $N = 9815$ individual cells. Based on the suggestions of [7] we use the expression levels of $D = 14$ genes that are relevant in understanding this biological process. The names of these set of 14 genes are listed in the Appendix C.7. We specifically demonstrate two different approaches of estimating the latent cellular ordering based on the same biological process. As our first approach, we consider two different gene expression levels to estimate latent cellular ordering [3, 41]. Specifically, we model the unspliced and spliced gene expressions [90] together using our developed pcHSGP to estimate the cellular ordering as latent inputs. As an alternative approach, we consider the spliced RNA gene expression levels and RNA velocity [47], a derivative information that is calculated based on the rate of change between spliced and unspliced gene expression levels. In this case, we use pdHSGP to simultaneously model spliced gene expression and their derivative RNA velocity to estimate the same latent cellular ordering. As a pre-processing step, we standardize the gene expression values (and RNA velocities) for both the approaches. This is done to overcome the high amount of variations among different gene (outputs) thus making it easy to specify priors on covariance function hyperparameters.

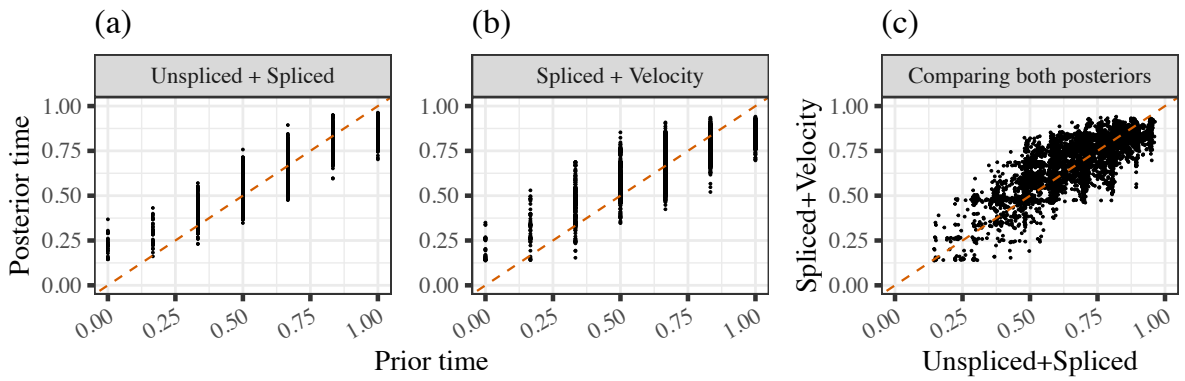

Figure 5.5: *Deviations of posterior latent cell ordering from discrete experimental times based on a) modeling unspliced and spliced gene expression using pcHSGP and b) modeling spliced gene expression and RNA velocity using pdHSGP. c) Comparison of posterior latent cell orderings from both the models.*

For the pcHSGP model (used in the first approach), we specify the priors for length-scales $\boldsymbol{\rho}_f$, $\boldsymbol{\rho}_g \sim \text{Normal}^+(0.3, 0.1^2)$, GP marginal SDs $\boldsymbol{\alpha}_f$, $\boldsymbol{\alpha}_g \sim \text{Normal}^+(0.5, 0.1^2)$, and error SDs $\boldsymbol{\sigma}_f$, $\boldsymbol{\sigma}_g \sim \text{Normal}^+(0.5, 0.1^2)$. The prior choices for the length-scales reflect similar levels of non-linearity in the functions $\mathbf{f}$ and $\mathbf{g}$ as in our simulation studies. The marginal and error SD priors are based on the standardized gene expression values. In case of the pdHSGP model, we use the same respective prior distributions for the set of hyperparameters length-scale $\boldsymbol{\rho}$, GP marginal SDs $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^{(1)}$ and error SDs $\boldsymbol{\sigma}$, $\boldsymbol{\sigma}^{(1)}$. We assume the experimental time $\tilde{\mathbf{x}}$ ranging between 0 and 1 as a noisy measurement around $\mathbf{x}$ with a measurement SD $s = 0.1$ (see Section 5.3.6). Given the range of $\tilde{\mathbf{x}}$ we use a prior of $\mathbf{x} \sim \text{Uniform}(0, 1)$.

The HSGPs are specified using the same boundary conditions as in our simulation studies (see Section 5.4.2). Choosing an adequate number of basis functions M depends on the range of inputs, the length-scale prior, and the boundary conditions [61, 72]. For both our approaches, we use $M = 10$ which is higher than the suggested number of basis functions for a SE covariance

function under our model setup. We erred on the side of caution with a larger M, since the relationship between the minimum number of basis functions for derivative covariance functions needs to be studied further.

We show the estimated cellular ordering using the pcHSGP as well as pdHSGP in Fig. 5.4. Our estimated cellular ordering exhibits a continuous temporal sequence based on each cells. From a biological perspective, this continuous nature of the ordering is crucial for understanding the trajectory of the biological process, as opposed to only a few discrete points [90]. We thus enable future research that better describes the underlying transition processes of blood progenitors to erythroid cells based on our estimated continuous cellular ordering. Secondly, the estimated posteriors shows strong deviations from the prior experimental time for both the approaches in Fig. 5.5(a) and (b). Combined with the evidence provided in our simulation studies, we attribute these deviations towards learning the true cellular ordering. Furthermore, in Fig. 5.5(c) we see that the estimated orderings from pcHSGP (analyzing unspliced and spliced gene expressions) and pdHSGP (analyzing spliced gene expressions and RNA velocity) deviate from one another indicating independent information to be learned about the biological process from either of the approaches.

5.6 Discussion

In this paper, we develop a scalable class of models for latent variable estimation based on approximate composite GPs, with a specific focus on approximating derivative GPs. The GP approximation is achieved by generalizing Hilbert space methods to obtain a reduced-rank representation of the composite covariance function through its spectral decomposition. Specifically, we derive and analyze the spectral decomposition of derivative covariance functions and further study their properties theoretically. Through our approximations, we reduce the steep cubic computational complexity of exact composite GPs (and derivative GPs) to linear with respect to the number of observations. Our method thus allows analyzing large data with composite and derivative GPs, which remains infeasible with similarly specified exact GPs. In our simulation studies, we show that the proposed GP approximations are superior compared to their exact GP counterparts, both in terms of speed and estimation accuracy for latent variable inputs.

We illustrate our models on a real-world case study depicting the maturation process of blood progenitors cells to erythroid cells. Our models promise a full reconstruction of the continuous temporal sequence of cells by refining a discrete quantity denoting only a few experimental time points. This continuous temporal ordering holds the potential to better explain the underlying biological process as compared to the experimental time. Using our models, we thus provide a strong method for estimating cellular ordering by analyzing full-sized single-cell RNA sequencing data. That said, our models can be easily applied to latent variable estimation problems arising from other fields of study as well.

5.6.1 Limitations and future research

Our composite and derivative GP approximations depend on the partial covariance structure. In this specification, we assume that the two GPs are independent, thus relaxing the off-diagonal functions in the joint covariance structure. We hypothesize that even more accurate latent variable estimates can be obtained by approximating the joint derivative GPs. However, this would require further generalizing the Hilbert space approximations to non-positive definite (and possibly asymmetric) kernel functions, which we consider out of scope of the present paper.

The latent variable estimation in our models require a set of noisy observations that are assumed to follow a Gaussian distribution centered around the true latent inputs, with a known measurement error (see Eq.(5.21)). Depending on the application scenario, this measurement error could be unknown. Thus, future studies should investigate this latent prior specification under unknown measurement error, as well as under entirely different distributional assumptions.

In our real-world case study, we present two approaches to estimate the cellular ordering – using unspliced and spliced gene expression data as well a combination of spliced gene expressions and its derivative RNA velocity. An in-depth comparison between these two approaches would require further case studies using various single-cell data, both real and simulated. Future research should investigate the advantages (or disadvantages) in choosing one approach over another in estimating the latent cellular ordering.

Code availability

The codes related to model development, simulation studies and the case study can be found here: `https://github.com/Soham6298/Latent-Composite-HSGPs`.

Acknowledgments

This work was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via the Collaborative Research Center 391 (Spatio-Temporal Statistics for the Transition of Energy and Transport) – 520388526 and DFG Project 500663361. Additionally, M.Z. was supported by the Else Kröner Fresenius Stiftung (ClinBrain). The authors acknowledge the computational resources provided by the German Network for Bioinformatics Infrastructure – de.NBI. The authors futher thank the International Max Planck Research School for Intelligent Systems for supporting S.M and M.Z. We are grateful to Luna Fazio for reading the manuscript and for thoughtful feedback that contributed to improving the paper.

Chapter 6

Conclusion

In this thesis, we develop a class of latent variable models using derivative Gaussian processes. Our models include several extensions of Gaussian processes (GPs) namely latent variable GPs, multi-output GPs as well and derivative GPs under a single framework and obtain highly accurate latent variable estimates. Our modified derivative covariance function tackle complexities in the data generating process like multi-dimensional output data, their derivative information and latent variable inputs. Furthermore, our models account for scale differences between outputs and their derivatives, varying information across multiple output dimensions as well as interactions between outputs, support for which standard GPs lack. Through our use of Bayesian inference, we provide uncertainty estimates for each latent variable sample. Evidenced by various experiments, our proposed methods increase latent variable estimation accuracy by including derivative information leveraging the joint covariance function modifications. Moreover, we demonstrate that without our proposed covariance structure, including the derivatives might yield misleading latent variable estimates, advocating the importance of our methods.

As with any exact GP model, our proposed methods face cubic and quadratic computational complexity with respect to sample size and number of output dimensions, respectively. We overcome the limited applicability for larger datasets by extending the state-of-the-art Hilbert space approximations for multi-output and latent input settings. Using the Hilbert space Gaussian process (HSGP) framework, we approximate the covariance function through a reduced-rank representation based on its spectral decomposition computed from a finite set of basis functions. This approximation method reduces the computational complexity to linear in both sample size and the number of basis functions, thus improving inference speed over their exact model counterparts. Compared to other GP approximation methods, the HSGPs prove to be better at posterior uncertainty calibration and estimation accuracy for latent variable samples under various simulated data scenarios.

To tackle our exact derivative GPs, we further extend the Hilbert space methods for the partial composite GPs, where we model a pair of data source as different outputs. We thus obtain a spectral approximation of the partial composite covariance functions using Hilbert space methods. Then, as a special case of partial composite GPs, we develop scalable partial derivative

GPs by modeling the outputs along with their derivatives. Specifically, we derive and analyze the spectral decomposition of derivative covariance functions and study their properties theoretically. Through the extended Hilbert space approximations on our modified derivative (and composite) covariance structure, we obtain a scalable class of derivative GPs, that can be widely applicable for latent variable estimation in large sample data scenarios.

Based on our methods, we demonstrate two approaches for estimating the unobserved cellular ordering in single-cell biology. First, where we analyze unspliced and spliced gene expression levels using the approximate composite GPs, and second, using approximate derivative GPs to analyze spliced expressions and its corresponding derivative RNA velocity. Through our methods, we reconstruct the continuous temporal latent ordering for each cell based on their gene expression levels as a step towards better understanding the underlying biological processes.

6.1 Future outlook

A limitation of our framework is in their stationarity assumption based on the choice of the covariance functions as well as the developed approximation methods. This limits their applicability to non-stationary data as evidenced by our simulation study of periodic data with an added non-linear trend shown in Chapter 3. While this is a general limitation of stationary GPs, the limitation currently lies in not having a derivative version of non-stationary covariance functions as well as their approximation method that focus on high quality latent variable estimation. An interesting future research would be to develop derivative GPs for non-stationary data where the primary focus would be on obtaining derivative versions of non-stationary covariance functions along with their approximate representations and verifying their performance for latent variable estimation.

Another aspect for future research is the choice of prior distributions. Here, we focused on informative priors for the GP hyperparameters in both our simulation studies and the real-world case study, although they are difficult to come by organically. Our methods will likely benefit from using stronger priors informed by the application-specific subject matter knowledge, specifically in sparse data scenarios. This not only applies to priors for the GP hyperparameters, but also to the priors of the latent input variables. Specifically, the latent variable estimation in our models require a set of noisy observations that are assumed to follow a Gaussian distribution centered around the true latent inputs, with a known measurement error. Depending on the application scenario, this measurement error could be unknown. Thus, future studies should investigate this latent prior specification under unknown measurement error, as well as under entirely different distributional assumptions. Moreover, a joint prior on the covariance function hyperparameters along with latent inputs will likely further improve model convergence. These future developments would foster a wider applicability of the proposed methods in this work, thereby further increasing their ability to accurately estimate latent variables.

When it comes to modeling multi-dimensional outputs, a natural consideration concerning the across-dimension correlation matrix $\mathbf{C}$ is whether to model it as input dependent or input independent. Throughout this work, we chose it to be input independent, thus estimating

the full correlation matrix directly. As an alternative, an input dependent formulation could be achieved through another GP that models the correlations across output dimensions within each observation. In this regard, it would further be interesting to study Hilbert space approximations also for this across-output dimension GP, especially when the number of dimensions is large enough for exact GPs to become computationally impractical.

The approximation for our derivative GPs depend on the partial covariance structure we propose in Chapter 5. In this specification, we assume that the two GPs are independent, thus relaxing the off-diagonal functions in the joint covariance structure. Based on the simulation experiments, we hypothesize that even more accurate latent variable estimates can be obtained by approximating the joint derivative GPs. However, this would require further generalizing the Hilbert space approximations to non-positive definite (and possibly asymmetric) kernel functions, which we believe is an important future prospect.

From an application perspective, in our real-world case studies, we present two approaches to estimate the cellular ordering – using unspliced and spliced gene expression data as well as a combination of spliced gene expressions and its derivative RNA velocity. To better suggest the practicality of the matter, an in-depth comparison between these two approaches would require further case studies using various single-cell data. This should involve both diverse real-world cases along with simulated data from biologically relevant simulation procedures. Thus, future research should investigate and directly compare the advantages in choosing one approach over another when it comes to estimating the latent cellular ordering.

References

[1] Abramowitz, M. and Stegun, I. A. (1965). *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*. Courier Corporation.

[2] Adler, R. J. (1981). *The Geometry of Random Fields*. Wiley, Chichester.

[3] Ahmed, S., Rattray, M., and Boukouvalas, A. (2019). GrandPrix: scaling up the Bayesian GPLVM for single-cell data. *Bioinformatics*, 35(1):47–54.

[4] Álvarez, M. and Lawrence, N. (2008). Sparse Convolved Gaussian Processes for Multi-output Regression. In *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.

[5] Álvarez, M., Luengo, D., and Lawrence, N. D. (2009). Latent Force Models. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, pages 9–16. PMLR. ISSN: 1938-7228.

[6] Álvarez, M. A. and Lawrence, N. D. (2011). Computationally Efficient Convolved Multiple Output Gaussian Processes. *Journal of Machine Learning Research*, 12(41):1459–1500.

[7] Barile, M., Imaz-Rosshandler, I., Inzani, I., Ghazanfar, S., Nichols, J., Marioni, J. C., Guibentif, C., and Göttgens, B. (2021). Coordinated changes in gene expression kinetics underlie both mouse and human erythroid maturation. *Genome Biology*, 22(1):197.

[8] Bauer, M., van der Wilk, M., and Rasmussen, C. E. (2016). Understanding Probabilistic Sparse Gaussian Process Approximations. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

[9] Bergen, V., Lange, M., Peidli, S., Wolf, F. A., and Theis, F. J. (2020). Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature Biotechnology*.

[10] Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. (2019). Pyro: deep universal probabilistic programming. *J. Mach. Learn. Res.*, 20(1):973–978.

[11] Bodin, E., Campbell, N. D. F., and Ek, C. H. (2017). Latent gaussian process regression.

[12] Bonilla, E. V., Chai, K., and Williams, C. (2007). Multi-task Gaussian Process Prediction. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc.

[13] Bourotte, M., Allard, D., and Porcu, E. (2016). A flexible class of non-separable cross-covariance functions for multivariate space–time data. *Spatial Statistics*, 18:125–146. Spatial Statistics Avignon: Emerging Patterns.

[14] Brooks, S., Gelman, A., Jones, G., and Meng, X.-L., editors (2011). *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, New York.

[15] Buettner, F., Natarajan, K. N., Casale, F. P., Proserpio, V., Scialdone, A., Theis, F. J., Teichmann, S. A., Marioni, J. C., and Stegle, O. (2015). Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology*, 33(2):155–160. Number: 2 Publisher: Nature Publishing Group.

[16] Buettner, F. and Theis, F. J. (2012). A novel approach for resolving differences in single-cell gene expression patterns from zygote to blastocyst. *Bioinformatics*, 28(18):i626–i632.

[17] Bürkner, P.-C. (2017). brms: An R Package for Bayesian Multilevel Models Using Stan. *Journal of Statistical Software*, 80:1–28.

[18] Bürkner, P.-C. (2018). Advanced bayesian multilevel modeling with the r package brms. *The R Journal*, 10:395–411.

[19] Bürkner, P.-C., Gabry, J., Kay, M., and Vehtari, A. (2023). posterior: Tools for working with posterior distributions. R package version 1.5.0.

[20] Campbell, K. and Yau, C. (2015). Bayesian Gaussian Process Latent Variable Models for pseudotime inference in single-cell RNA-seq data. Pages: 026872 Section: New Results.

[21] Campbell, K. R. and Yau, C. (2018). A descriptive marker gene approach to single-cell pseudotime inference. *Bioinformatics*, 35(1):28–35.

[22] Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32.

[23] Chatfield, C. (2003). *The Analysis of Time Series*, chapter 6, pages 92–104. Chapman and Hall/CRC.

[24] Courant, R. and Hilbert, D. (2008). *Methods of Mathematical Physics, vol. 1*. Wiley, Hoboken.

[25] Cressie, N. A. C. (1993). *Statistics for Spatial Data*. Wiley, New York.

[26] Csató, L. and Opper, M. (2002). Sparse on-line gaussian processes. *Neural Computation*, 14(3):641–668.

[27] Eriksson, D., Dong, K., Lee, E. H., Bindel, D., and Wilson, A. G. (2018). Scaling Gaussian Process Regression with Derivatives. arXiv:1810.12283 [cs, stat].

[28] Evans, L. (2010). *Partial Differential Equations*. Graduate studies in mathematics. American Mathematical Society.

[29] Fazio, L., Scholz, M., Aguilar, J. E., and Bürkner, P.-C. (2025). Primed priors for simulation-based validation of bayesian models.

[30] Folland, G. B. (1995). *Introduction to Partial Differential Equations: Second Edition*, volume 102. Princeton University Press, ned - new edition edition.

[31] Gihman, I. I. and Skorokhod, A. V. (2004). *Linear Theory of Random Processes*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[32] Gneiting, T., Kleiber, W., and Schlather, M. (2010). Matérn cross-covariance functions for multivariate random fields. *Journal of the American Statistical Association*, 105(491):1167–1177.

[33] Golub, G. H. and Van Loan, C. F. (1989). *Matrix Computations*. Johns Hopkins University Press, Baltimore. Second edition.

[34] Goovaerts, P. (1997). *Geostatistics For Natural Resources Evaluation*. Oxford University Press, USA.

[35] Gorin, G., Fang, M., Chari, T., and Pachter, L. (2022). Rna velocity unraveled. *PLoS Comput Biol 18(9): e1010492.*

[36] Gorsuch, R. (1983). *Factor Analysis (2nd ed.).* Psychology Press.

[37] Gupta, R., Cerletti, D., Gut, G., Oxenius, A., and Claassen, M. (2022). Simulation-based inference of differentiation trajectories from RNA velocity fields. *Cell Reports Methods*, 2(12):100359.

[38] Guttorp, P. and Gneiting, T. (2006). On the Whittle-Matérn correlation family. *Biometrika, Volume 93, Issue 4.*

[39] Haque, A., Engel, J., Teichmann, S. A., and Lönnberg, T. (2017). A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications. *Genome Medicine*, 9(1):75.

[40] Hensman, J., Fusi, N., and Lawrence, N. D. (2013a). Gaussian Processes for Big Data. arXiv:1309.6835 [cs, stat].

[41] Hensman, J., Lawrence, N. D., and Rattray, M. (2013b). Hierarchical Bayesian modelling of gene expression time series across irregularly sampled replicates and clusters. *BMC Bioinformatics*, 14(1):252.

[42] Hoffman, D. M. and Gelman, A. (2014). The no-u-turn sampler. *Journal of Machine Learning Research.*

[43] Joukov, V. and Kulić, D. (2022). Fast Approximate Multioutput Gaussian Processes. *IEEE Intelligent Systems*, 37(4):56–69. Conference Name: IEEE Intelligent Systems.

[44] Journel, A. G. and Huijbregts, C. J. (1978). *Mining Geostatistics*. Academic Press, London.

[45] Kline, R. B. (1998). *Principles and practice of structural equation modeling*. Guilford Press.

[46] Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. (2017). Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(14):1–45.

[47] La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M. E., Lönnerberg, P., Furlan, A., Fan, J., Borm, L. E., Liu, Z., van Bruggen, D., Guo, J., He, X., Barker, R., Sundström, E., Castelo-Branco, G., Cramer, P., Adameyko, I., Linnarsson, S., and Kharchenko, P. V. (2018). RNA velocity of single cells. *Nature*, 560(7719):494–498.

[48] Lalchand, V., Ravuri, A., and Lawrence, N. D. (2022). Generalised GPLVM with Stochastic Variational Inference. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, pages 7841–7864. PMLR. ISSN: 2640-3498.

[49] Lawrence, N. (2003). Gaussian Process Latent Variable Models for Visualisation of High Dimensional Data. In *Advances in Neural Information Processing Systems*, volume 16. MIT Press.

[50] Lawrence, N. (2005). Probabilistic non-linear principal component analysis with gaussian process latent variable models. *Journal of Machine Learning Research*, 6(60):1783–1816.

[51] Lazarsfeld, P. F. and Henry, N. W. (1968). *Latent structure analysis*. Houghton Mifflin, Boston.

[52] Lewandowski, D., Kurowicka, D., and Joe, H. (2009). Generating random correlation matrices based on vines and extended onion method. *Journal of Multivariate Analysis*, 100(9):1989–2001.

[53] Loehlin, J. and Beaujean, A. (2016). *Latent Variable Models: An Introduction to Factor, Path, and Structural Equation Analysis, Fifth Edition (5th ed.)*. Routledge.

[54] Losick, R. and Desplan, C. (2008). Stochasticity and Cell Fate. *Science*, 320(5872):65–68. Publisher: American Association for the Advancement of Science.

[55] Maamar, H., Raj, A., and Dubnau, D. (2007). Noise in Gene Expression Determines Cell Fate in Bacillus subtilis. *Science*, 317(5837):526–529. Publisher: American Association for the Advancement of Science.

[56] Mahdessian, D., Cesnik, A. J., Gnann, C., Danielsson, F., Stenström, L., Arif, M., Zhang, C., Le, T., Johansson, F., Schutten, R., Bäckström, A., Axelsson, U., Thul, P., Cho, N. H., Carja, O., Uhlén, M., Mardinoglu, A., Stadler, C., Lindskog, C., Ayoglu, B., Leonetti, M. D., Pontén, F., Sullivan, D. P., and Lundberg, E. (2021). Spatiotemporal dissection of the cell cycle with single-cell proteogenomics. *Nature*, 590(7847):649–654. Number: 7847 Publisher: Nature Publishing Group.

[57] Mercer, J. (1909). Xvi. functions of positive and negative type, and their connection the theory of integral equations. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 209(441-458):415–446.

[58] Modrák, M., Moon, A. H., Kim, S., Bürkner, P., Huurre, N., Faltejsková, K., Gelman, A., and Vehtari, A. (2023). Simulation-Based Calibration Checking for Bayesian Computation: The Choice of Test Quantities Shapes Sensitivity. *arXiv preprint*.

[59] Moreno-Muñoz, P., Artés, A., and Álvarez, M. (2018). Heterogeneous Multi-output Gaussian Process Prediction. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

[60] Mukherjee, S., Claassen, M., and Bürkner, P.-C. (2025a). DGP-LVM: Derivative Gaussian process latent variable models. *Statistics and Computing*, 35(5):120.

[61] Mukherjee, S., Claassen, M., and Bürkner, P.-C. (2025b). Hilbert space methods for approximating multi-output latent variable gaussian processes. *arXiv preprint arXiv:2505.16919*.

[62] Neal, R. (2011). *"MCMC Using Hamiltonian Dynamics." In Handbook of Markov Chain Monte Carlo, edited by Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao-Li Meng, 116–62.* Chapman; Hall/CRC.

[63] O'Hagan, A. (1978). Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society: Series B (Methodological)*, 40(1):1–24.

[64] O'hagan, A. (1992). Some bayesian numerical analysis. In *Bayesian Statistics 4: Proceedings of the Fourth Valencia International Meeting, Dedicated to the memory of Morris H. DeGroot, 1931–1989*. Oxford University Press.

[65] Padidar, M., Zhu, X., Huang, L., Gardner, J., and Bindel, D. (2021). Scaling Gaussian Processes with Derivative Information Using Variational Inference. In *Advances in Neural Information Processing Systems*, volume 34, pages 6442–6453. Curran Associates, Inc.

[66] Pijuan-Sala, B., Griffiths, J. A., Guibentif, C., Hiscock, T. W., Jawaid, W., Calero-Nieto, F. J., Mulas, C., Ibarra-Soria, X., Tyser, R. C. V., Ho, D. L. L., Reik, W., Srinivas, S., Simons, B. D., Nichols, J., Marioni, J. C., and Göttgens, B. (2019). A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature*, 566(7745):490–495.

[67] Quinonero-Candela, J. and Rasmussen, C. E. (2005). A unifying view of sparse approximate gaussian process regression. *Journal of machine learning research*, 6(Dec):1939–1959.

[68] Raj, A. and Oudenaarden, A. v. (2008). Nature, Nurture, or Chance: Stochastic Gene Expression and Its Consequences. *Cell*, 135(2):216–226. Publisher: Elsevier.

[69] Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

[70] Reid, J. E. and Wernisch, L. (2016). Pseudotime estimation: deconfounding single cell time series. *Bioinformatics*, 32(19):2973–2980.

[71] Riba, A., Oravecz, A., Durik, M., Jiménez, S., Alunni, V., Cerciat, M., Jung, M., Keime, C., Keyes, W. M., and Molina, N. (2022). Cell cycle gene regulation dynamics revealed by rna velocity and deep-learning. *Nature Communications*, 13(1):2865.

[72] Riutort-Mayol, G., Bürkner, P.-C., Andersen, M. R., Solin, A., and Vehtari, A. (2022). Practical Hilbert space approximate Bayesian Gaussian processes for probabilistic programming. *Statistics and Computing*, 33(1).

[73] Seeger, M. W., Williams, C. K. I., and Lawrence, N. D. (2003). Fast Forward Selection to Speed Up Sparse Gaussian Process Regression. In *International Workshop on Artificial Intelligence and Statistics*, pages 254–261. PMLR. ISSN: 2640-3498.

[74] Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society: Series B (Methodological)*, 47(1):1–21.

[75] Smola, A. and Bartlett, P. (2000). Sparse Greedy Gaussian Process Regression. In *Advances in Neural Information Processing Systems*, volume 13. MIT Press.

[76] Snelson, E. and Ghahramani, Z. (2005). Sparse Gaussian Processes using Pseudo-inputs. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press.

[77] Solak, E., Murray-smith, R., Leithead, W., Leith, D., and Rasmussen, C. (2002). Derivative Observations in Gaussian Process Models of Dynamic Systems. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press.

[78] Solin, A. and Särkkä, S. (2020). Hilbert space methods for reduced-rank Gaussian process regression. *Statistics and Computing*, 30(2):419–446.

[79] Stan Development Team (2023). RStan: the R interface to Stan. R package version 2.21.8.

[80] Stan Development Team (2024). *Stan Modeling Language Users Guide and Reference Manual, 2.32.0.*

[81] Stein, E. and Shakarchi, R. (2011). *Fourier Analysis: An Introduction*. Princeton lectures in analysis. Princeton University Press.

[82] Stein, M. L. (1999). *Interpolation of Spatial Data*. Springer Series in Statistics. Springer, New York, NY.

[83] Säilynoja, T., Bürkner, P.-C., and Vehtari, A. (2022). Graphical test for discrete uniformity and its applications in goodness-of-fit evaluation and multiple sample comparison. *Statistics and Computing*, 32(2):32.

[84] Talts, S., Betancourt, M., Simpson, D., Vehtari, A., and Gelman, A. (2020). Validating Bayesian Inference Algorithms with Simulation-Based Calibration. *arXiv preprint*.

[85] Teh, Y. W., Seeger, M., and Jordan, M. I. (2005). Semiparametric Latent Factor Models. *Proceedings of Machine Learning Research*.

[86] Tipping, M. E. and Bishop, C. M. (2002). Probabilistic principal component analysis. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 61(3):611–622.

[87] Titsias, M. (2009). Variational Learning of Inducing Variables in Sparse Gaussian Processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, pages 567–574. PMLR. ISSN: 1938-7228.

[88] Titsias, M. and Lawrence, N. D. (2010). Bayesian Gaussian Process Latent Variable Model. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 844–851. JMLR Workshop and Conference Proceedings. ISSN: 1938-7228.

[89] Trangucci, R., Betancourt, M., and Vehtari, A. (2016). Prior formulation for gaussian process hyperparameters. *Practical Bayesian Nonparameterics Workshop - NeurIPS*.

[90] Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., Lennon, N. J., Livak, K. J., Mikkelsen, T. S., and Rinn, J. L. (2014). The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology*, 32(4):381–386. Number: 4 Publisher: Nature Publishing Group.

[91] Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., and Bürkner, P.-C. (2021). Rank-Normalization, Folding, and Localization: An Improved $\widehat{R}$ for Assessing Convergence of MCMC (with Discussion). *Bayesian Analysis*, 16(2):667 – 718.

[92] Wall, M. M. (2004). A close look at the spatial structure implied by the car and sar models. *Journal of statistical planning and inference*, 121(2):311–324.

[93] Wiener, N. (1964). *Extrapolation, interpolation, and smoothing of stationary time series*. The MIT press.

[94] Williams, C. and Rasmussen, C. (1995a). Gaussian processes for regression. In Touretzky, D., Mozer, M., and Hasselmo, M., editors, *Advances in Neural Information Processing Systems*, volume 8. MIT Press.

[95] Williams, C. and Rasmussen, C. (1995b). Gaussian Processes for Regression. In *Advances in Neural Information Processing Systems*, volume 8. MIT Press.

[96] Wood, S. N. (2003). Thin-plate regression splines. *Journal of the Royal Statistical Society (B)*, 65(1):95–114.

[97] Yao, Y., Vehtari, A., Simpson, D., and Gelman, A. (2018). Yes, but did it work?: Evaluating variational inference. In *International Conference on Machine Learning*, pages 5581–5590. PMLR.

[98] Yue, H. and Chan, K. S. (1996). Asymptotic efficiency of the sample mean in markov chain monte carlo schemes. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(3):525–539.

Appendix A

Declaration

This appendix is for Chapter 3 and is based on the appendix of the following published manuscript:

> *DGP-LVM: Derivative Gaussian process latent variable models*
> **Soham Mukherjee**, Manfred Claassen and Paul-Christian Bürkner
> Stat Comput 35, 120 – Published 8 June 2025
> https://doi.org/10.1007/s11222-025-10644-4

Text, figures and tables are adapted from the appendix of manuscript https://arxiv.org/abs/2404.04074

A.1: Derivative covariance functions

Here we show the mathematical details of the general covariance function structure as well as specific derivative forms of SE, Matérn 3/2 and Matérn 5/2 covariance functions.

Proof of the derivative covariance function structure

Lemma: Let $X \in \mathcal{X}$ be a random variable and $g : \mathbb{R} \times \mathcal{X} \to \mathbb{R}$ is a function $\ni g(t, X)$ is integrable $\forall t$ and g is continuously differentiable w.r.t t. Assume a random variable $Z \ni \left|\frac{\delta}{\delta t} g(t, X)\right| \leq Z$ almost surely $\forall t$ and $E(Z) < \infty$, then $\frac{\delta}{\delta t} E(g(t, X)) = E\left(\frac{\delta}{\delta t} g(t, X)\right)$.

Let us consider y_i and v_j such that $v_j = \frac{\delta y_j}{\delta x_j}$ where we consider a Gaussian Process model with $y_i = f(x_i) + \epsilon_i$. For a GP model $\text{Cov}(y_i, y_j)$ is completely defined using corresponding inputs

x_i and x_j by a covariance function. We see that

$$\begin{aligned}
\text{Cov}(y_i, v_j) &= \mathbb{E}\left((y_i - \mathbb{E}\left(y_i\right))\left(v_j - \mathbb{E}(v_j)\right)\right) \\
&= \mathbb{E}\left(\left(y_i - \mathbb{E}\left(y_i\right)\right)\left(\frac{\delta}{\delta x_j} y_j - \mathbb{E}\left(\frac{\delta}{\delta x_j} y_j\right)\right)\right) \\
&= \mathbb{E}\left(\left(y_i - \mathbb{E}\left(y_i\right)\right)\left(\frac{\delta}{\delta x_j} y_j - \frac{\delta}{\delta x_j}\left(\mathbb{E}\left(y_j\right)\right)\right)\right) && \text{(By DCT)} \\
&= \mathbb{E}\left(\left(y_i - \mathbb{E}\left(y_i\right)\right)\frac{\delta}{\delta x_j}\left(y_j - \left(\mathbb{E}\left(y_j\right)\right)\right)\right) && \text{(derivative over subtraction)} \\
&= \mathbb{E}\left(\frac{\delta}{\delta x_j}\left(y_i - \mathbb{E}\left(y_i\right)\right)\left(y_j - \left(\mathbb{E}\left(y_j\right)\right)\right)\right) && (y_i\text{'s are constant w.r.t } x_j) \\
&= \frac{\delta}{\delta x_j}\left(\mathbb{E}\left(\left(y_i - \mathbb{E}\left(y_i\right)\right)\left(y_j - \left(\mathbb{E}\left(y_j\right)\right)\right)\right)\right) && \text{(By DCT)} \\
&= \frac{\delta}{\delta x_j}\text{Cov}(y_i, y_j).
\end{aligned}$$

Using similar reasoning and with $v_i = \frac{\delta y_i}{\delta x_j}$ and $v_j = \frac{\delta y_j}{\delta x_j}$, we find

$$\begin{aligned}
\text{Cov}(v_i, v_j) &= \mathbb{E}\left(\left(v_i - \mathbb{E}\left(v_i\right)\right)\left(v_j - \mathbb{E}\left(v_j\right)\right)\right) \\
&= \mathbb{E}\left(\left(\frac{\delta}{\delta x_i} y_i - \mathbb{E}\left(\frac{\delta}{\delta x_i} y_i\right)\right)\left(\frac{\delta}{\delta x_j} y_j - \mathbb{E}\left(\frac{\delta}{\delta x_j} y_j\right)\right)\right) \\
&= \mathbb{E}\left(\left(\frac{\delta}{\delta x_i} y_i - \frac{\delta}{\delta x_i}\mathbb{E}\left(y_i\right)\right)\left(\frac{\delta}{\delta x_j} y_j - \frac{\delta}{\delta x_j}\mathbb{E}\left(y_j\right)\right)\right) && \text{(By DCT)} \\
&= \mathbb{E}\left(\frac{\delta}{\delta x_i}\left(\frac{\delta}{\delta x_j}\left(y_i - \mathbb{E}\left(y_i\right)\right)\left(y_j - \mathbb{E}\left(y_j\right)\right)\right)\right) && \text{(derivative over constants)} \\
&= \frac{\delta^2}{\delta x_i \delta x_j}\text{Cov}(y_i, y_j) && \text{(By DCT)}
\end{aligned}$$

Derivative covariance functions

In the following covariance functions, α is GP marginal SD corresponding to output $\mathbf{y}$; $\alpha^{(1)}$ is the GP marginal SD corresponding to derivative output $\mathbf{y}^{(1)}$; ρ is the GP length scale parameter.

1. **Squared Exponential**

$$k = \alpha^2 \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right)$$

$$k^{(0,1)} = \frac{\delta k}{\delta x_j} = \alpha\alpha^{(1)} \frac{(x_i - x_j)}{\rho^2} \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right)$$

$$k^{(1,0)} = \frac{\delta k}{\delta x_i} = \alpha\alpha^{(1)} \frac{(x_j - x_i)}{\rho^2} \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right)$$

$$k^{(1,1)} = \frac{\delta^2 k}{\delta x_i \delta x_j} = \frac{{\alpha^{(1)}}^2}{\rho^4}(\rho^2 - (x_i - x_j)^2) \exp\left(-\frac{(x_i - x_j)^2}{2\rho^2}\right)$$

2. **Matérn 3/2**

$$k = \alpha^2 \left(1 + \frac{\sqrt{3(x_i - x_j)^2}}{\rho}\right) \exp\left(-\frac{\sqrt{3(x_i - x_j)^2}}{\rho}\right)$$

$$k^{(0,1)} = \frac{\delta k}{\delta x_j} = \alpha\alpha^{(1)} \left(\frac{3(x_i - x_j)}{\rho^2}\right) \exp\left(-\frac{\sqrt{3(x_i - x_j)^2}}{\rho}\right)$$

$$k^{(1,0)} = \frac{\delta k}{\delta x_i} = \alpha\alpha^{(1)} \left(\frac{3(x_j - x_i)}{\rho^2}\right) \exp\left(-\frac{\sqrt{3(x_i - x_j)^2}}{\rho}\right)$$

$$k^{(1,1)} = \frac{\delta^2 k}{\delta x_i \delta x_j} = {\alpha^{(1)}}^2 \left(\frac{3}{\rho^2}\right) \left(1 - \frac{\sqrt{3(x_i - x_j)^2}}{\rho}\right) \exp\left(-\frac{\sqrt{3(x_i - x_j)^2}}{\rho}\right)$$

3. **Matérn 5/2**

$$k = \alpha^2 \left(1 + \frac{\sqrt{5(x_i - x_j)^2}}{\rho} + \frac{5(x_i - x_j)^2}{3\rho^2}\right) \exp\left(-\frac{\sqrt{5(x_i - x_j)^2}}{\rho}\right)$$

$$k^{(0,1)} = \frac{\delta k}{\delta x_j} = \alpha\alpha^{(1)} \left(\frac{5(x_i - x_j)}{3\rho^2}\right) \left(1 + \frac{\sqrt{5(x_i - x_j)^2}}{\rho}\right) \exp\left(-\frac{\sqrt{5(x_i - x_j)^2}}{\rho}\right)$$

$$k^{(1,0)} = \frac{\delta k}{\delta x_i} = \alpha\alpha^{(1)} \left(\frac{5(x_j - x_i)}{3\rho^2}\right) \left(1 + \frac{\sqrt{5(x_i - x_j)^2}}{\rho}\right) \exp\left(-\frac{\sqrt{5(x_i - x_j)^2}}{\rho}\right)$$

$$k^{(1,1)} = \frac{\delta^2 k}{\delta x_i \delta x_j} = {\alpha^{(1)}}^2 \left(\frac{5}{3\rho^2}\right) \left(1 + \frac{\sqrt{5(x_i - x_j)^2}}{\rho} - \frac{5(x_i - x_j)^2}{\rho^2}\right) \exp\left(-\frac{\sqrt{5(x_i - x_j)^2}}{\rho}\right)$$

A.2: Additional simulation results

Here we show the additional plots from our simulation studies. Specifically, we provide the MCMC convergence diagnostics for Matérn 3/2, 5/2 as well as the periodic simulation scenarios. We then present the full versions of the model evaluation plots for recovery of true latent inputs $\mathbf{x}$ using RMSE and MAE as evaluation metrics for all the simulation scenarios. Further we show additional hyperparameter recovery plots based on enabling/disabling different model assumptions. Finally, we present the case where we use four MCMC chains for a reduced SE data simulation scenario that shows minimal or no effect on model evaluation metrics irrespective of number of MCMC chains.

Additional MCMC diagnostic plots

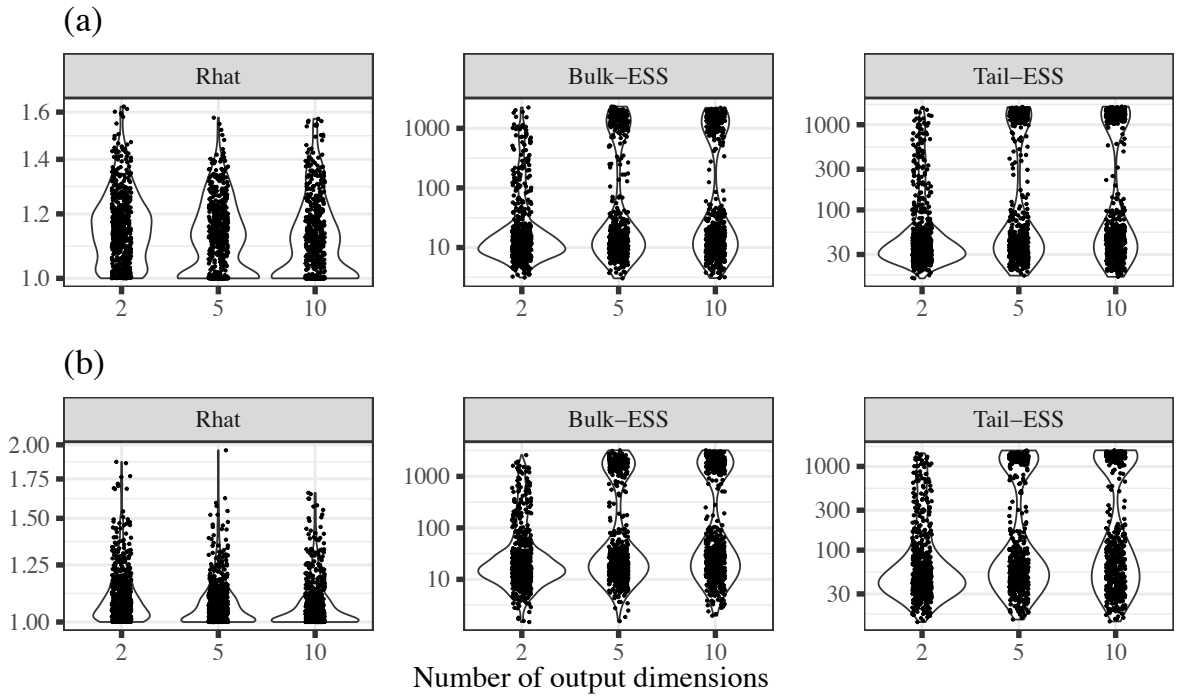

Figure A1: *Matérn 3/2 scenario: Convergence measures for (a) latent inputs and (b) GP hyperparameters. The individual points correspond to each fitted models per simulated data. The y-axis for Bulk and Tail ESS plots are log10 transformed.*

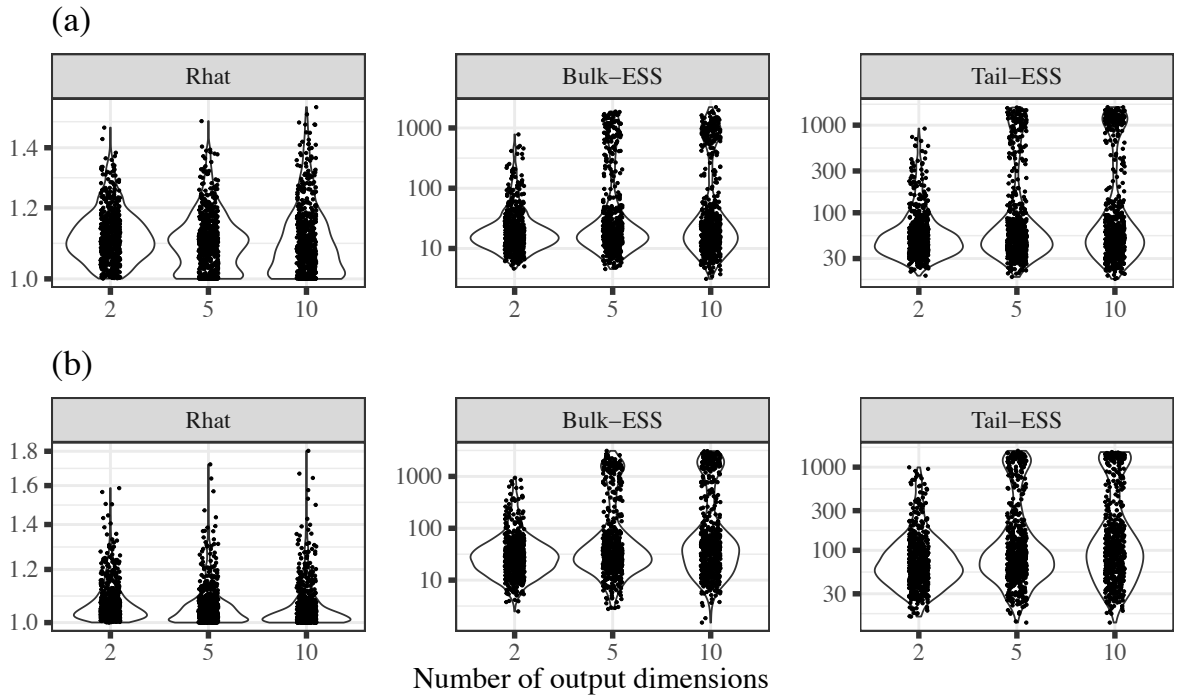

Figure A2: *Matérn 5/2 scenario: Convergence measures for (a) latent inputs and (b) GP hyperparameters. The individual points correspond to each fitted models per simulated data. The y-axis for Bulk and Tail ESS plots are log10 transformed.*

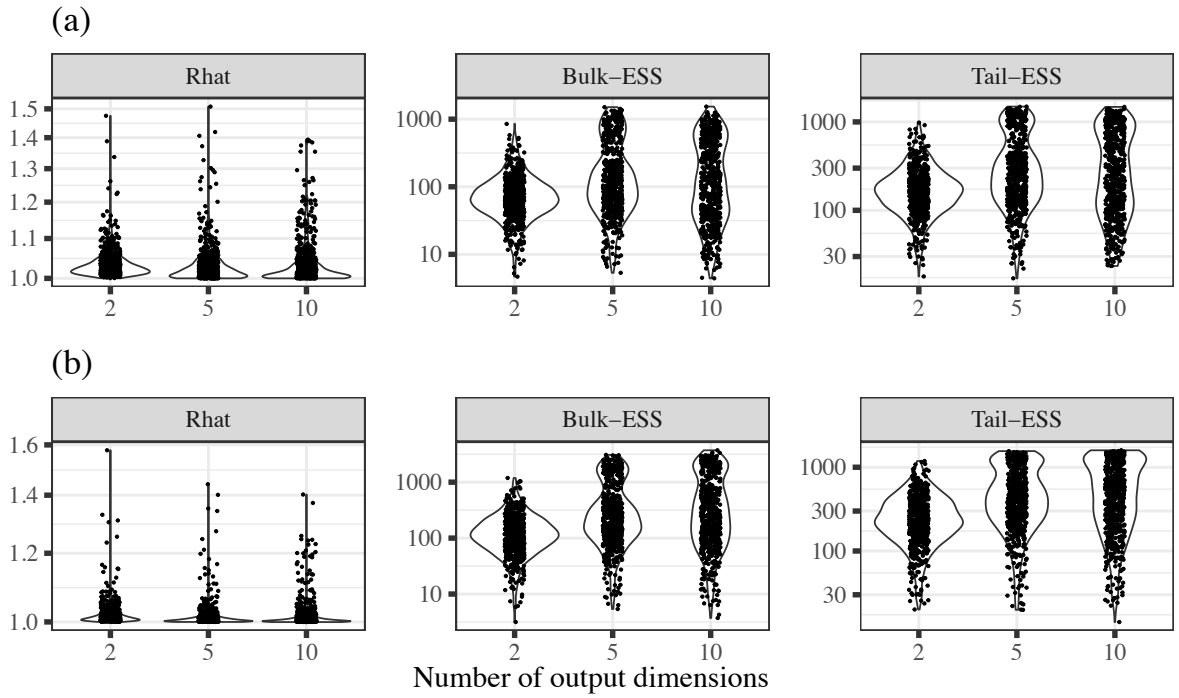

Figure A3: *Periodic scenario: Convergence measures for (a) latent inputs and (b) GP hyperparameters. The individual points correspond to each fitted models per simulated data. The y-axis for Bulk and Tail ESS plots are log10 transformed.*

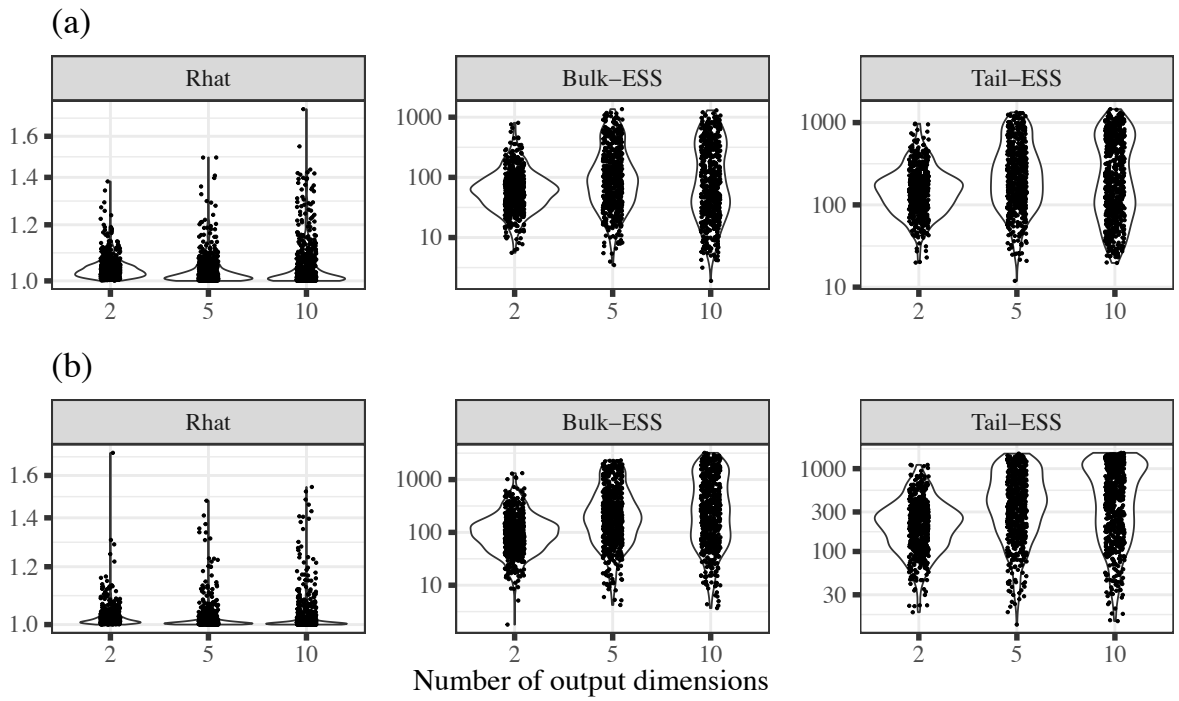

Figure A4: *Periodic with trend scenario: Convergence measures for (a) latent inputs and (b) GP hyperparameters. The individual points correspond to each fitted models per simulated data. The y-axis for Bulk and Tail ESS plots are log10 transformed.*

Full versions of model evaluation plots using RMSE: Latent inputs

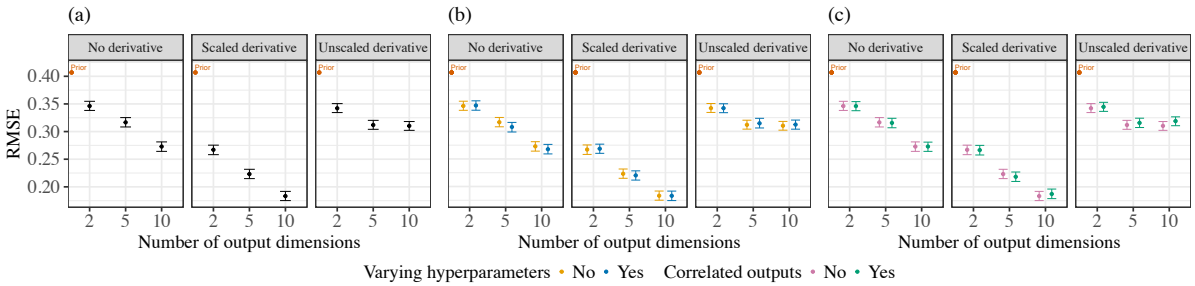

Figure A5: *Squared exponential scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using RMSE.*

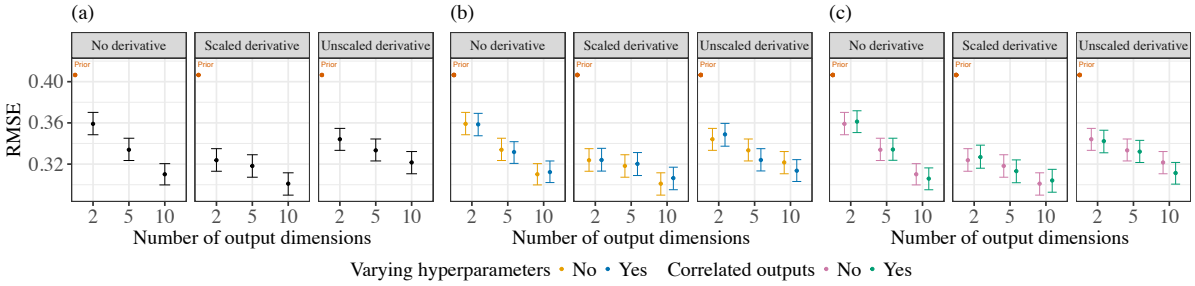

Figure A6: *Matérn 3/2 scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using RMSE.*

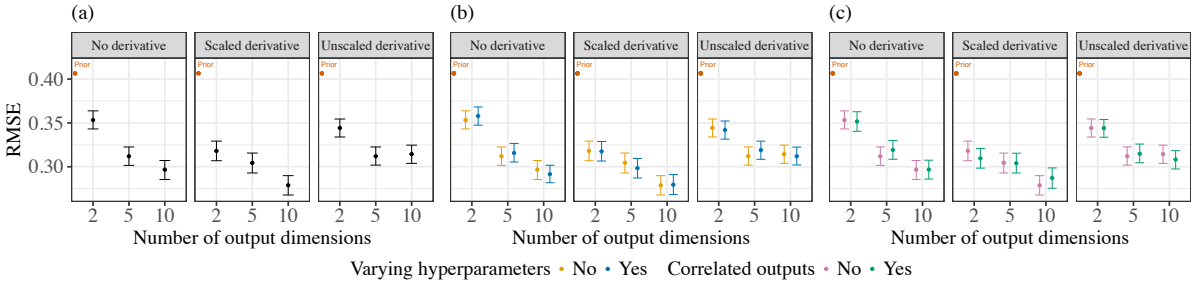

Figure A7: *Matérn 5/2 scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using RMSE.*

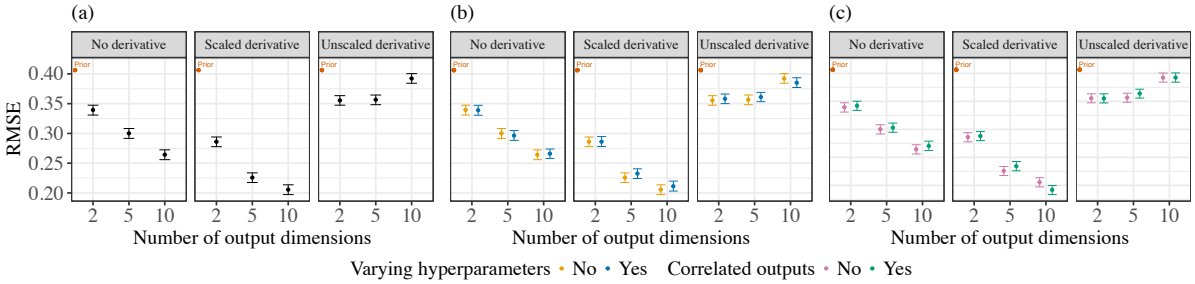

Figure A8: *Periodic scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using RMSE.*

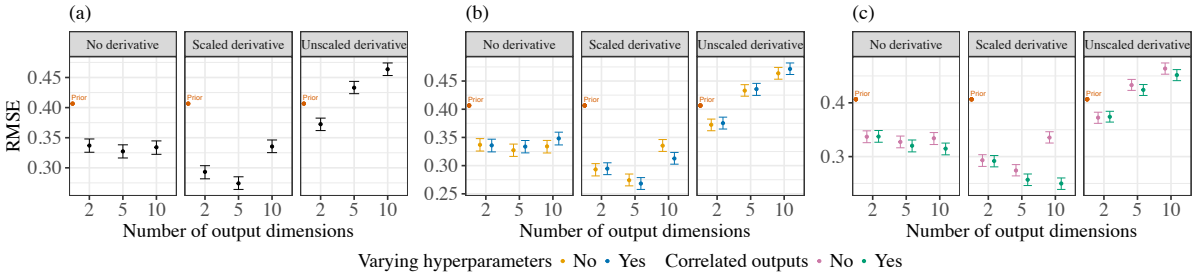

Figure A9: *Periodic with trend scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using RMSE.*

Full versions of model evaluation plots using MAE: Latent inputs

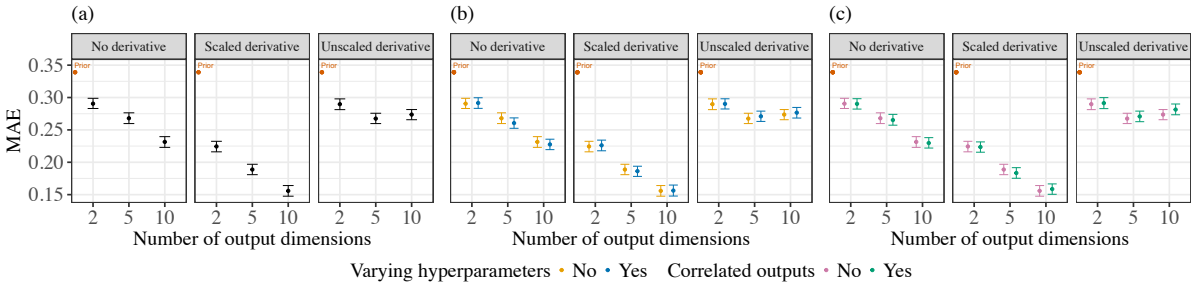

Figure A10: *Squared exponential scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using MAE.*

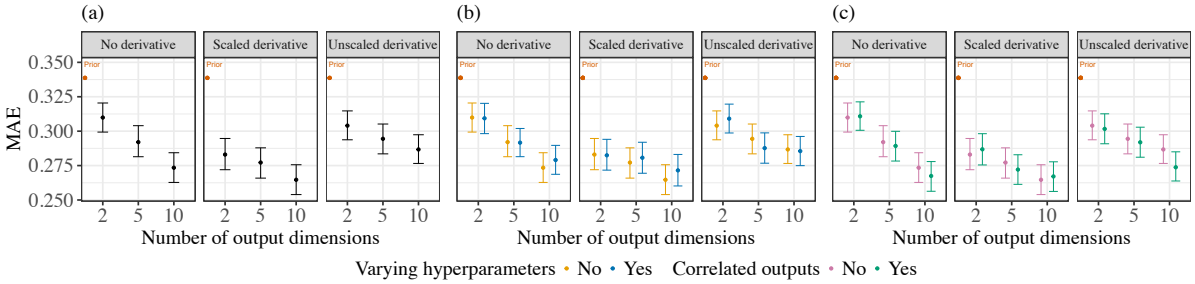

Figure A11: *Matérn 3/2 scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using MAE.*

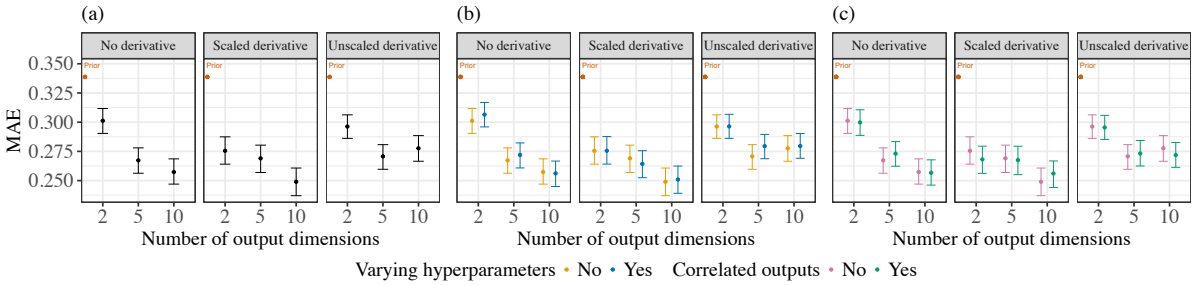

Figure A12: *Matérn 5/2 scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using MAE.*

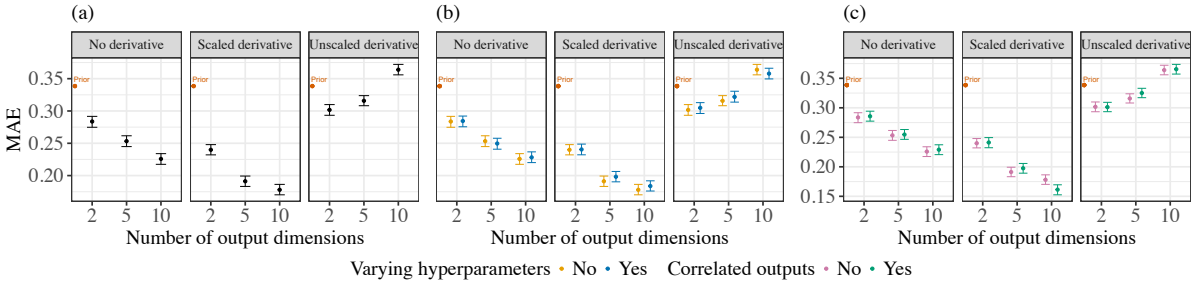

Figure A13: *Periodic scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using MAE.*

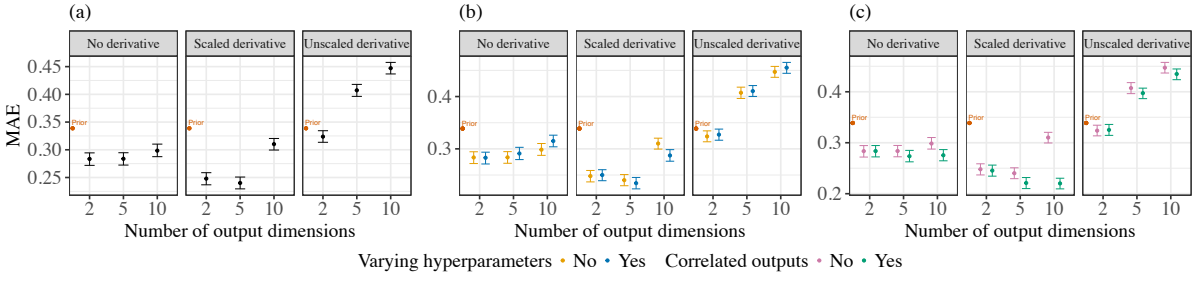

Figure A14: *Periodic with trend scenario: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using MAE.*

Additional model evaluation plots using RMSE:hyperparameters

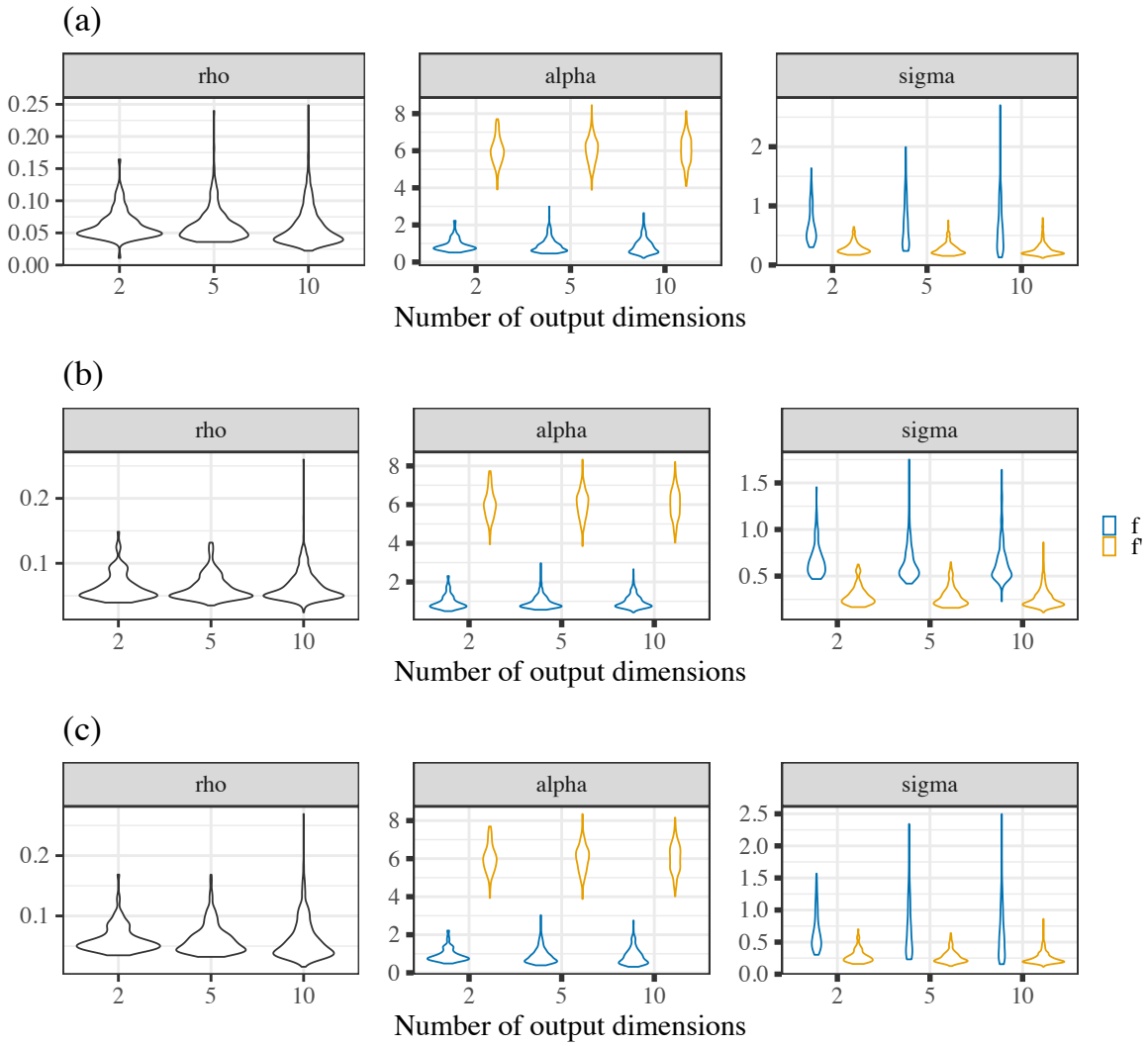

Figure A15: *Squared exponential scenario: Hyperparameter RMSEs for scaled derivatives (a) without varying hyperparameters, (b) without correlated outputs and (c) without both varying hyperparameters and correlated outputs. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

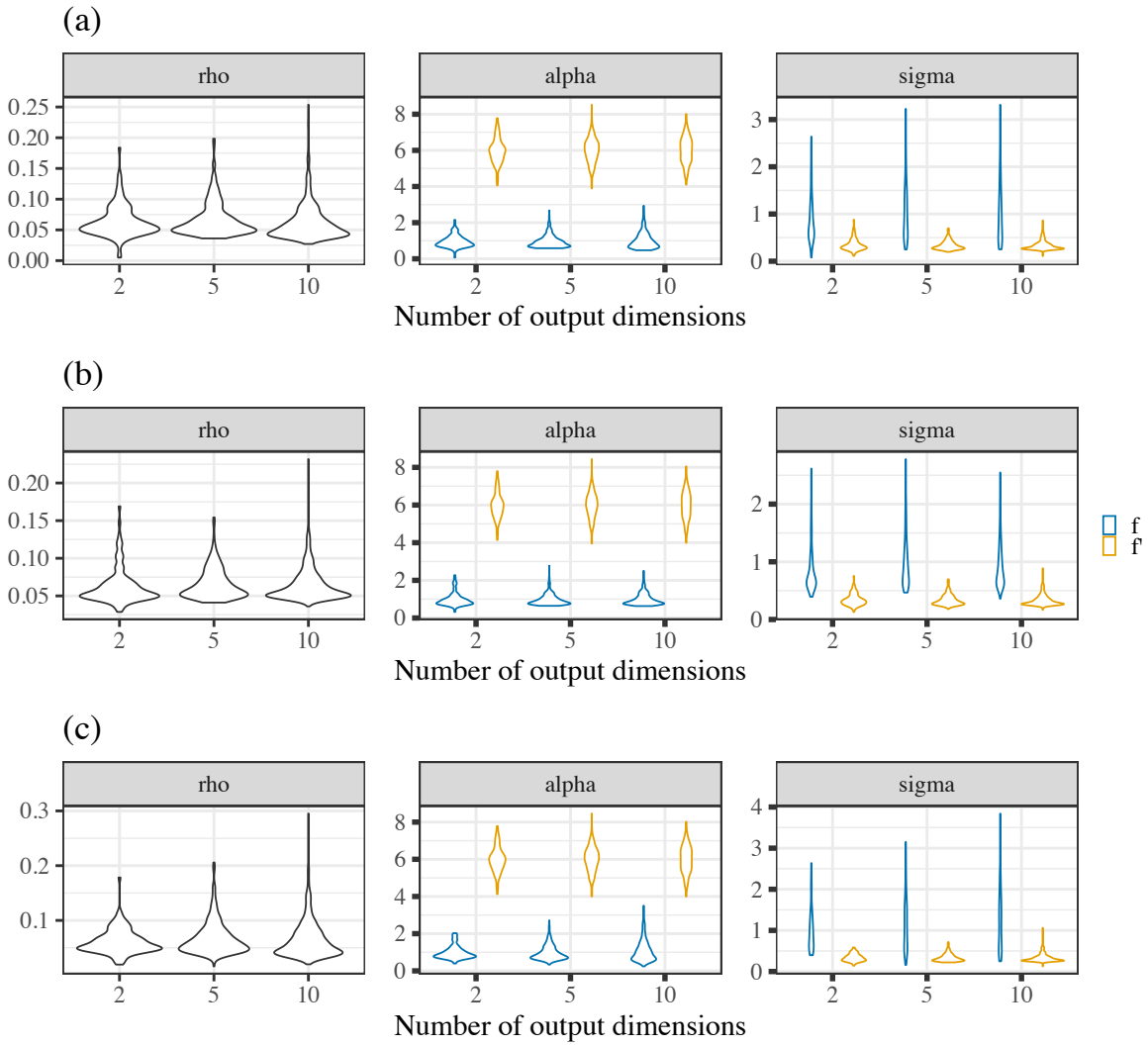

Figure A16: *Matérn 3/2 scenario: Hyperparameter RMSEs for scaled derivatives (a) without varying hyperparameters, (b) without correlated outputs and (c) without both varying hyperparameters and correlated outputs. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

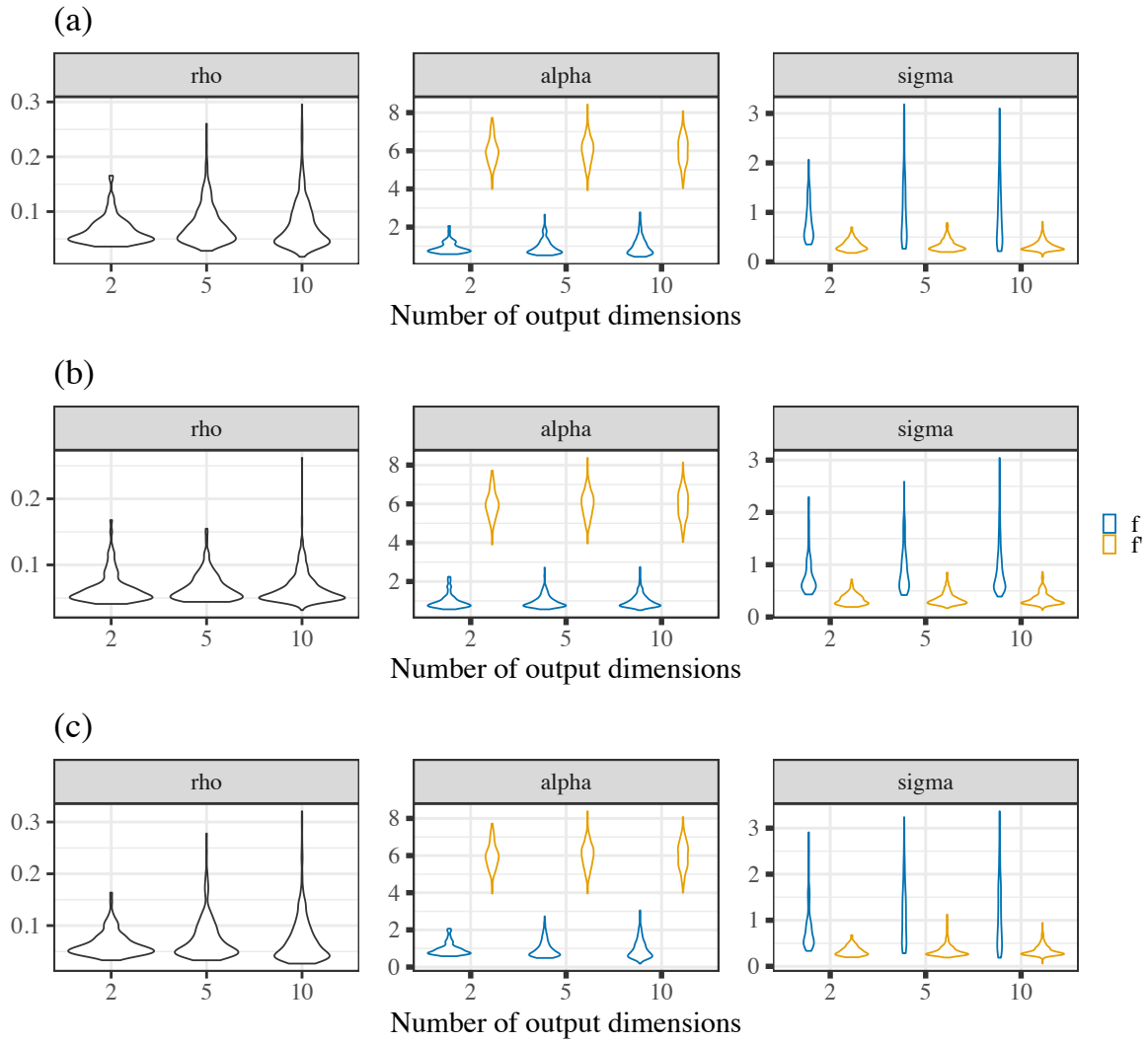

Figure A17: *Matérn 5/2 scenario: Hyperparameter RMSEs for scaled derivatives (a) without varying hyperparameters, (b) without correlated outputs and (c) without both varying hyperparameters and correlated outputs. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

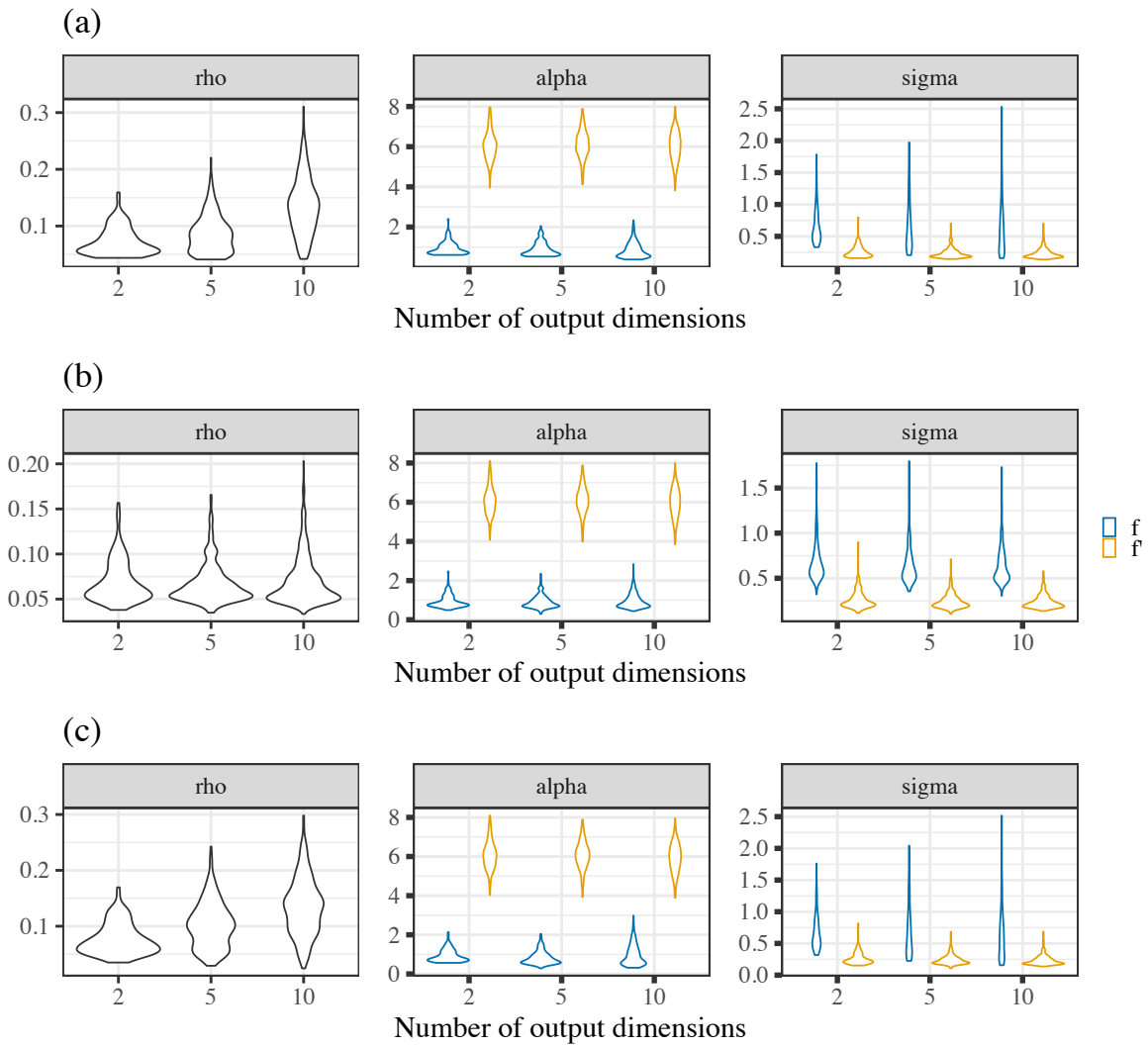

Figure A18: *Periodic scenario: Hyperparameter RMSEs for scaled derivatives (a) without varying hyperparameters, (b) without correlated outputs and (c) without both varying hyperparameters and correlated outputs. The different color denotes if the hyperparameters correspond to the original or the derivative part of the model.*

Model evaluation plot from simulation study with four MCMC chains

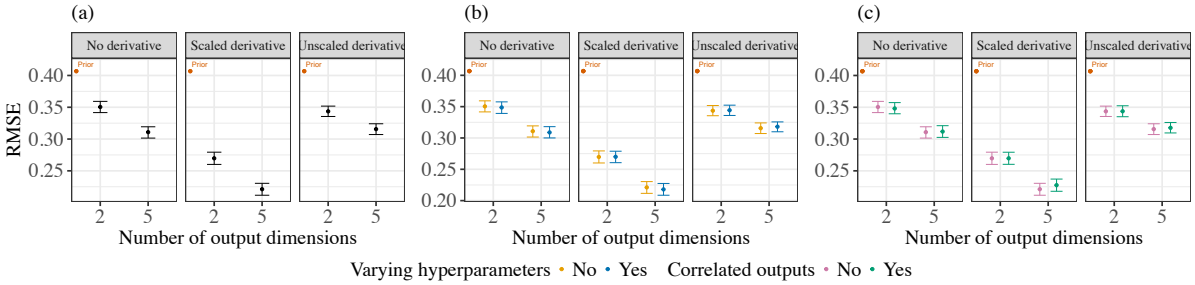

Figure A19: *Squared exponential scenario with 4 MCMC chains: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using RMSE.*

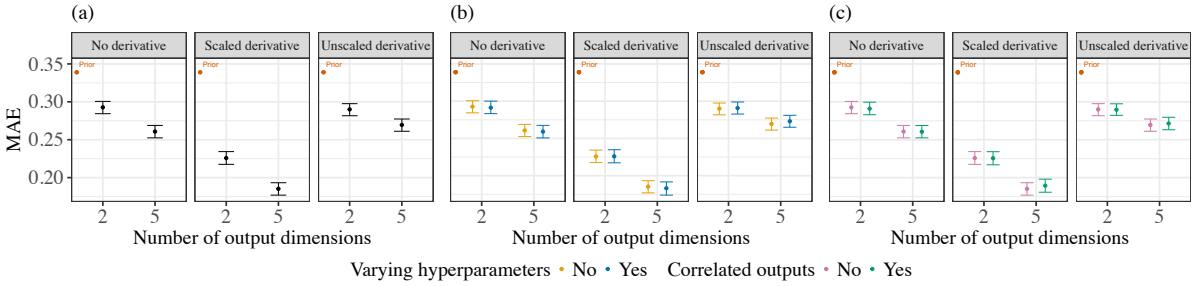

Figure A20: *Squared exponential scenario with 4 MCMC chains: main effects of including (a) scaled derivatives and interaction effects of assuming (b) varying hyperparameters and (c) correlated outputs on recovery of latent inputs (full version) using MAE.*

Appendix B

Declaration

This appendix is for Chapter 4 and is based on the supplementary materials of the following manuscript that is under review:

> *Hilbert space methods for approximating multi-output latent variable Gaussian processes*
> **Soham Mukherjee**, Manfred Claassen and Paul-Christian Bürkner
> arXiv preprint arXiv:2505.16919 – in review (2025)
> https://doi.org/10.48550/arXiv.2505.16919

Text, figures and tables are adapted from the supplementary materials of the manuscript https://arxiv.org/abs/2505.16919 with minor updates in notations.

B.1: Methods related to simulation study

Summarizing results

To evaluate the accuracy of estimating latent variables, we compare posterior samples of the latent input variable $\mathbf{x}$ denoted by $\mathbf{x}_{post}$ from each model with their respective ground truth values denoted by $\mathbf{x}_{true}$. Using $\text{Bias}(\mathbf{x}_{post}, \mathbf{x}_{true})^2$ and $\text{SD}(\mathbf{x}_{post})$, we look at the bias-SD trade-off in estimating the latent variables. We compute the posterior bias and SD from all fitted models for each sample size ($N = 20,\ 50$ and 200) and output dimension ($D = 5,\ 10$ and 20). We prefer models that provide both low bias indicating posterior mean estimates close to the ground truth as well as lower posterior SD indicating high precision. The reliability of posterior SD estimates depend on our models being well calibrated.

To analyze the results from our experiments, we use a multilevel analysis of variance model (ANOVA) fitted with brms [17], which disentangles the various components of our simulation study design. With $\boldsymbol{\mu}_{resp}$ and $\boldsymbol{\sigma}_{resp}$ being the mean and SD of our response variable, we use

$$\begin{aligned} \boldsymbol{\mu}_{resp} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{b} + \sum_{\tau} s_{\mu_\tau}(t_\tau) \\ \boldsymbol{\sigma}_{resp} &= \mathbf{X}\boldsymbol{\eta} + \mathbf{Z}\mathbf{u} + \sum_{\tau} s_{\sigma_\tau}(t_\tau) \end{aligned} \tag{6.1}$$

where $\boldsymbol{\beta},\ \mathbf{b}$ (for $\boldsymbol{\mu}_{resp}$) and $\boldsymbol{\eta},\ \mathbf{u}$ (for $\boldsymbol{\sigma}_{resp}$) are coefficients at the population and group levels

with $\mathbf{X}$ and $\mathbf{Z}$ being the corresponding design matrices. The $s_\tau(t_\tau)$ terms denote smooth functions over covariates t fitted via splines. We use these models to summarize the results from our experiments. In the simulation studies, the population level design matrix $\mathbf{X}$ contains covariates representing the approximation method and number of output dimensions along with their interaction terms. Based on our experimental setup, the approximation methods include the exact model, HSGPs (with three choices of basis functions) and VIGPs, thus creating a five level factor variable. We use a three level factor variable depicting the 5, 10, and 20 output dimensions. Through $\mathbf{Z}$ we account for the group-level dependency structure in the response induced by fitting multiple models to the same simulated dataset. We include a random intercept over datasets as well as corresponding random slopes of the approximation method choices. Lastly, to capture the non-linear relation of the response to the ground truth t, we introduce thin-plate regression spline terms $s_{\mu_\tau}(t_\tau)$ and $s_{\sigma_\tau}(t_\tau)$. The spline terms accounts for any non-linear relationship of the response with respect to the true parameter values t. We use the log γ scores as our response to summarize the model calibration results. To summarize the results of latent variable estimation, we subsequently specify posterior bias and SD as the response.

SBC

Using simulation-based calibration (SBC), we test model calibration for estimating latent inputs $\mathbf{x}$. We start with a model, say, $\mathcal{M}_0$. We then generate J datasets $\mathbf{y}^{(j)}, j = 1, \ldots, J$ each of size N from the data generating process that exactly aligns with the model $\mathcal{M}_0$. In other words, each individual dataset $\mathbf{y}^{(j)}$ is generated based on a corresponding model parameter draw $\mathbf{x}_0^{(j)}$ from its prior distribution $p(\mathbf{x})$. We sample from the posterior approximation by fitting $\mathcal{M}_0$ to each of the datasets $\mathbf{y}^{(j)}$ thus resulting in J fitted models $\mathcal{M}^{(j)}$ with respective posteriors $p(\mathbf{x} \mid \mathbf{y}^{(j)})$ each having H posterior draws $\mathbf{x}^{(j,h)}$. Using $\mathbf{x}_0^{(j)}$ as the ground truth, we then calculate a rank statistic for each univariate posterior quantity $\mathbf{h}_p(x)$ for a specific parameter by counting the number of posterior draws $\mathbf{h}_p(x^{(j,h)})$ that are smaller than $\mathbf{h}_p(x_0^{(j)})$. The rank statistic $R^{(j)}$ for the model $T^{(j)}$ is then given as

$$R^{(j)} = \sum_{h=1}^{H} \mathbb{I}[\mathbf{h}_p(x^{(j,h)}) < \mathbf{h}_p(x_0^{(j)})] \tag{6.2}$$

The distribution of these single rank-value per model taken together across all J models is a discrete uniform distribution if the approximate posteriors correspond to the true posteriors. Using this property, we assess the correctness of the posterior approximations by testing the rank distribution for uniformity. If the rank distribution departs from uniformity, it indicates a problem in the data generating process, model implementation, the posterior approximations or a combination of these.

We check SBC by computing the confidence bands for the empirical cumulative distribution function (ECDF) of the rank distribution and visualizing it under the assumption of the uniformity. An alternative, however is based on the probability of γ observing the most extreme

point on the ECDF under the assumption of uniformity. The test statistic is given by

$$\gamma = 2 \min_{j \in \{1,\ldots,J+1\}} (\min\{\text{Bin}(R_j \mid J, z_j), 1 - \text{Bin}(R_j - 1 \mid J, z_j)\}), \tag{6.3}$$

where z_j is the expected proportion of ranks below j such that $z_j = \frac{j}{J+1}$, R_j is the actual empirical ranks below j, $\text{Bin}(R_j \mid J, z_j)$ is the CDF of the Binomial distribution with J trials and the probability of success evaluated at R. The calculated γ scores (presented on the logarithmic scale) are then compared to a threshold value at which to reject uniformity. The log γ scores are advantageous in summarizing large number of parameters, different models as well us various simulation conditions. Thus, we prefer evaluating model calibration using the log γ scores in our case.

B.2: GP hyperparameters estimation

We show the estimation accuracy for hyperparameters of the exact GPs and HSGPs for all the simulation scenarios.The VIGP doesn't allow varying hyperparameter estimation for different output dimensions. Moreover, model inference using VI only provides a posterior point estimate for each of the hyperparameters and are thus excluded from this part of model evaluation comparison. We present the model evaluation summary using RMSE which combines the model-specific effects on posterior bias and SD.

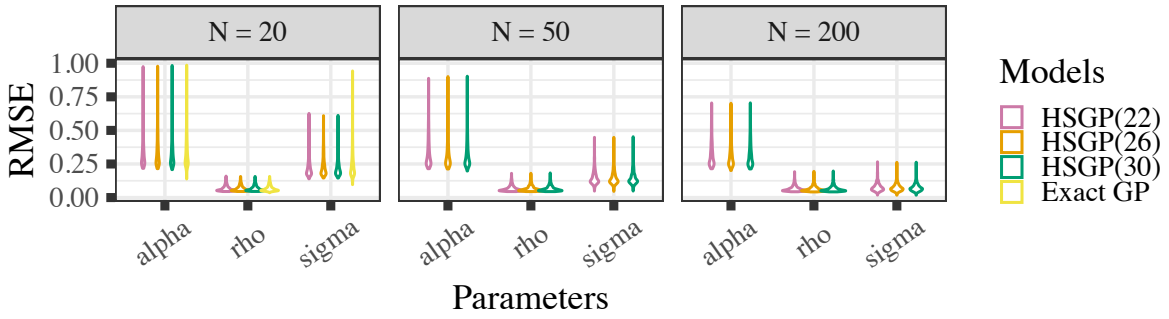

Figure B1: *Squared exponential scenario: RMSE on recovery of GP hyperparameters for exact GP and HSGP fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

We present the RMSE for each model hyperparameter under the SE data scenario in Figure B1 where we see how the decreasing effect on the RMSE of GP marginal SD $\boldsymbol{\alpha}$ and error SD $\boldsymbol{\sigma}$ as the sample size N increases. Since the results were qualitatively same for all the different choices of output dimensions D, we decided to showcase only for the $D = 20$. Interestingly, RMSE for the length-scale $\boldsymbol{\rho}$ stays similar across all sample size choices. We can thus be sure about the reliability of HSGPs in recovering true hyperparameter values for all kinds of data scenarios. In the more challenging case of the highly variable length-scale, we see in Figure B2 that the RMSE for the length-scale $\boldsymbol{\rho}$ increases as expected, but is still comparatively better than for the exact GP for $N = 20$. All the other simulation study scenarios are presented as

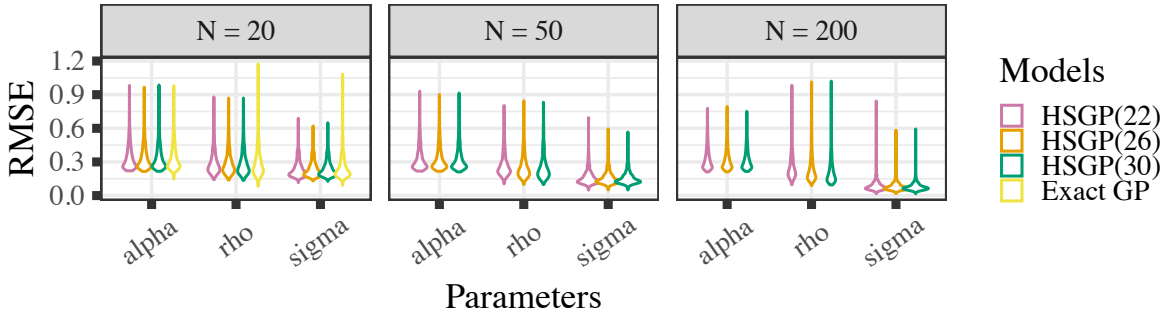

Figure B2: *Squared exponential scenario (highly varying ρ): RMSE on recovery of GP hyperparameters for exact GP and HSGP fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

additional plots in the Appendix B.6 (see Figures B14-B17).

B.3: MCMC convergence diagnostics (additional figures)

We present the model convergence diagnostics of exact GPs and HSGPs for all the simulation scenarios except for the squared exponential (SE) scenario (which is shown in the main manuscript). In all of the scenarios, the HSGPs consistently the 1.01 threshold of model convergence as suggested by [91]. HSGPs subsequently also have much higher Bulk and Tail-ESS. Based on the diagnostics, HSGPs show much more consistent and stable convergence as compared to exact GPs.

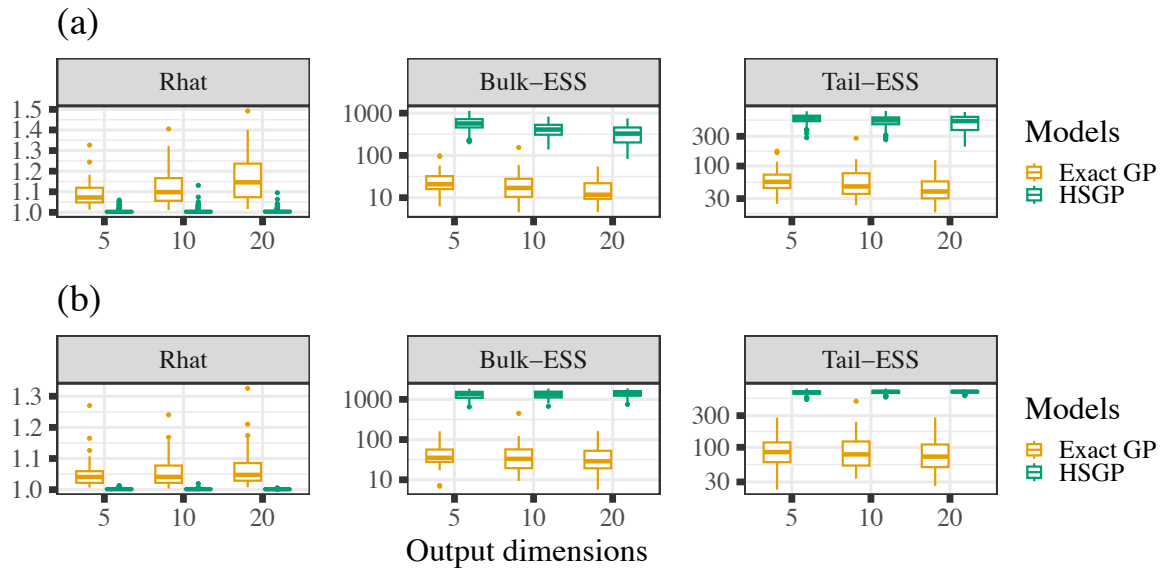

Figure B3: *Matérn 3/2 scenario: Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

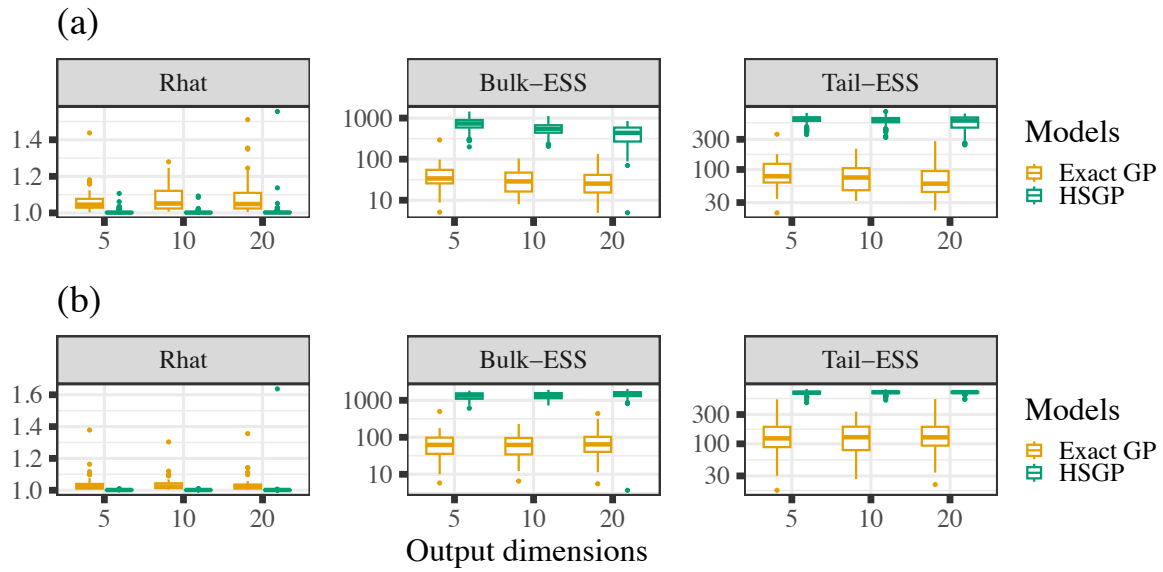

Figure B4: *Matérn 5/2 scenario: Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

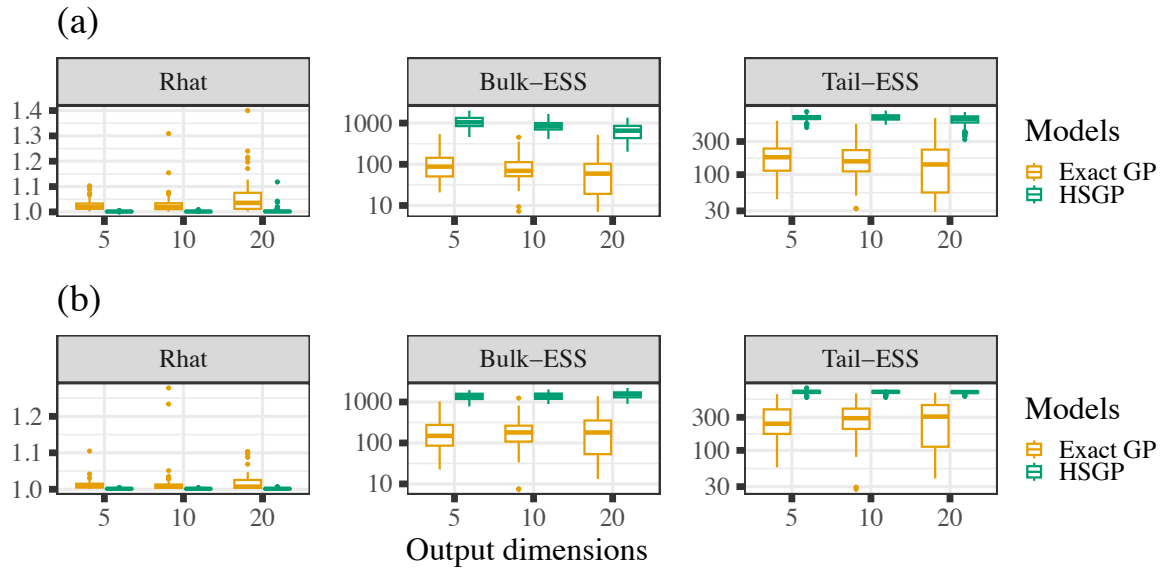

Figure B5: *Periodic data scenario (lower oscillation): Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

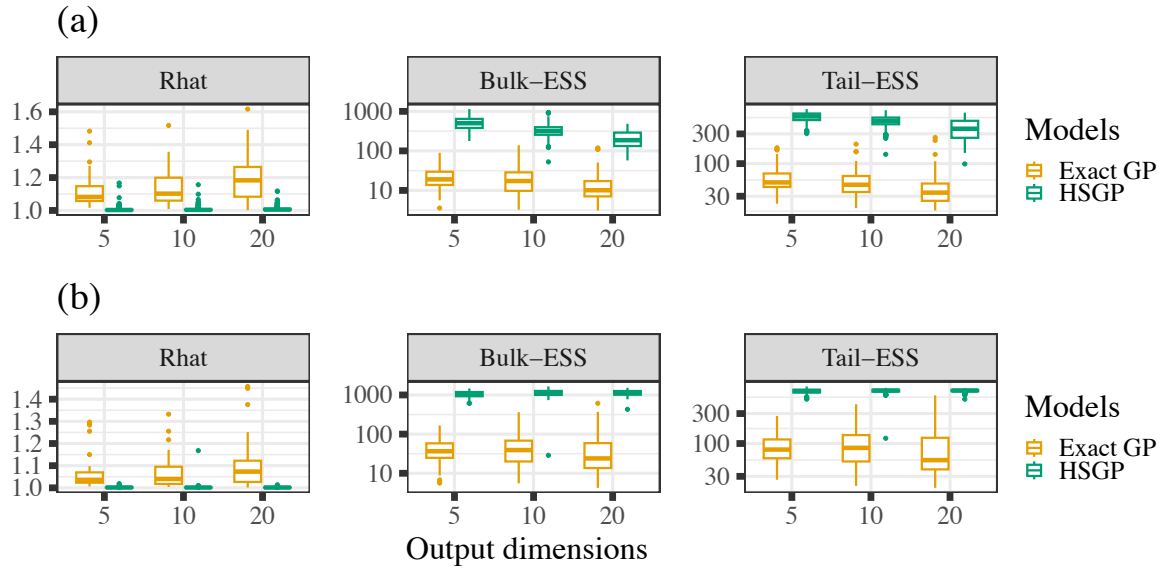

Figure B6: *Periodic data scenario (higher oscillation): Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

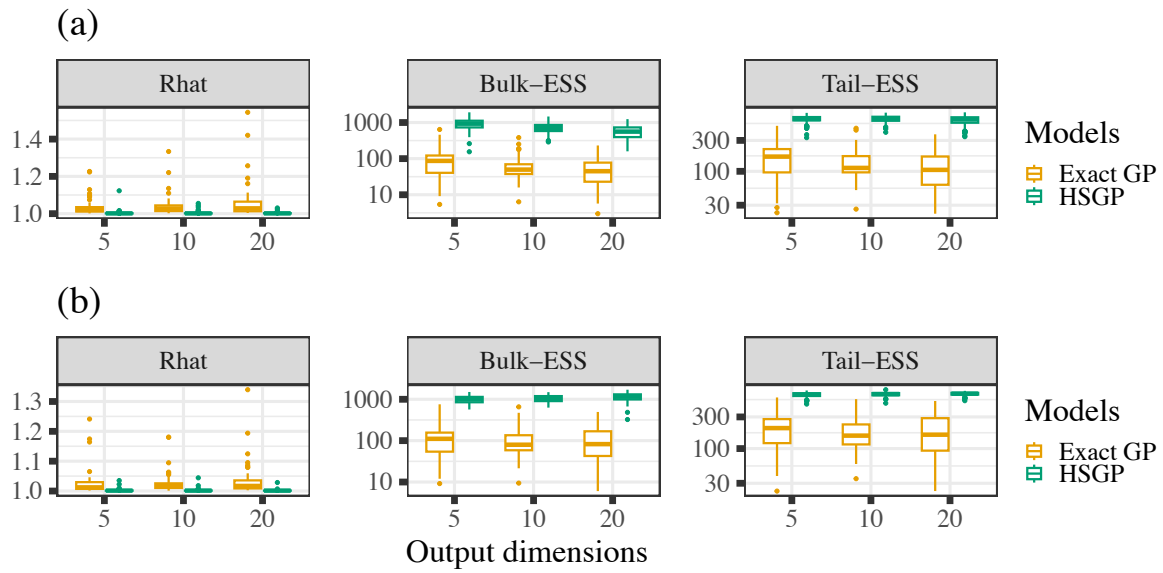

Figure B7: *Squared exponential scenario (highly varying ρ): Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

B.4: Model calibration tests (additional figures)

We present the log γ scores (offset by the 95% confidence threshold) for all the fitted models in the Matérn 3/2 and 5/2 simulation scenarios. The exact GPs and VIGPs consistently fail the calibration test for $D = 20$. The HSGPs with $M = 51$ number of representative basis functions show good calibration across all the scenarios.

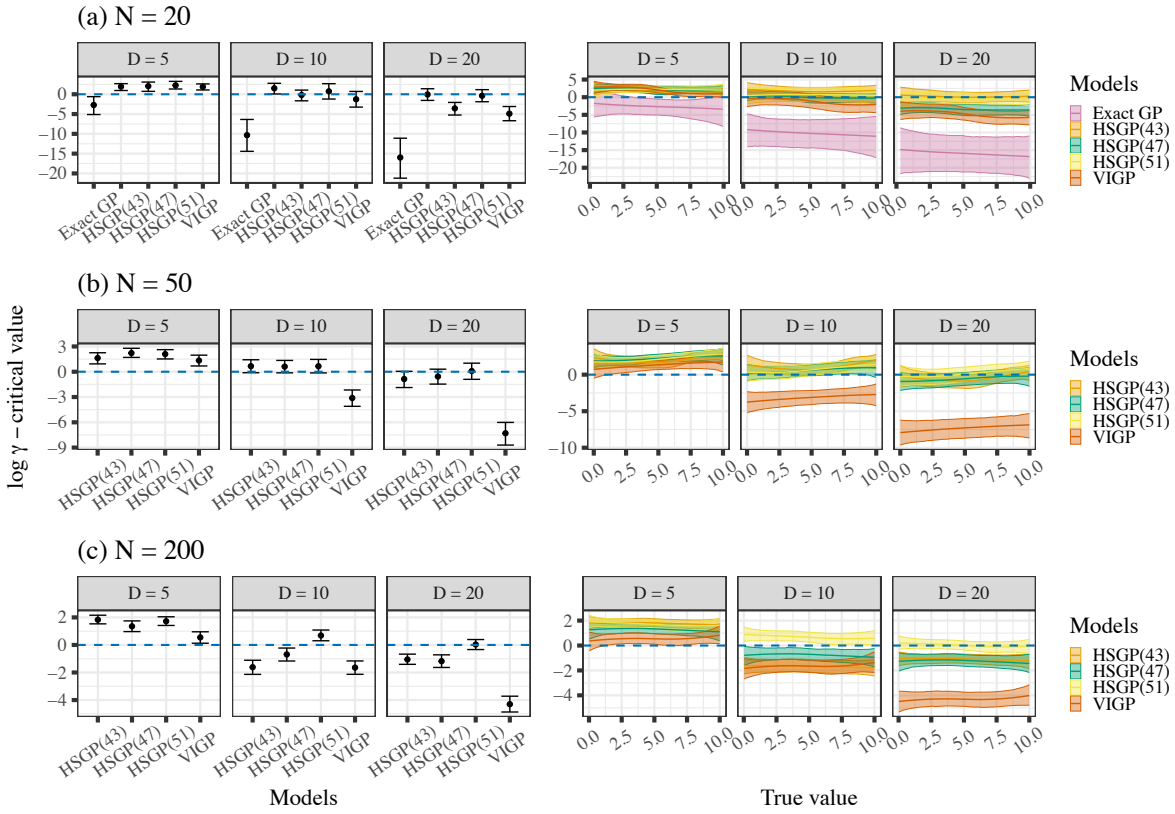

Figure B8: *Matérn 3/2 scenario: log γ scores offset by the 95% confidence threshold for all the fitted models. The behavior of scores across true latent x values are shown in the right-hand panel. The blue dashed line denotes the threshold to reject uniformity. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

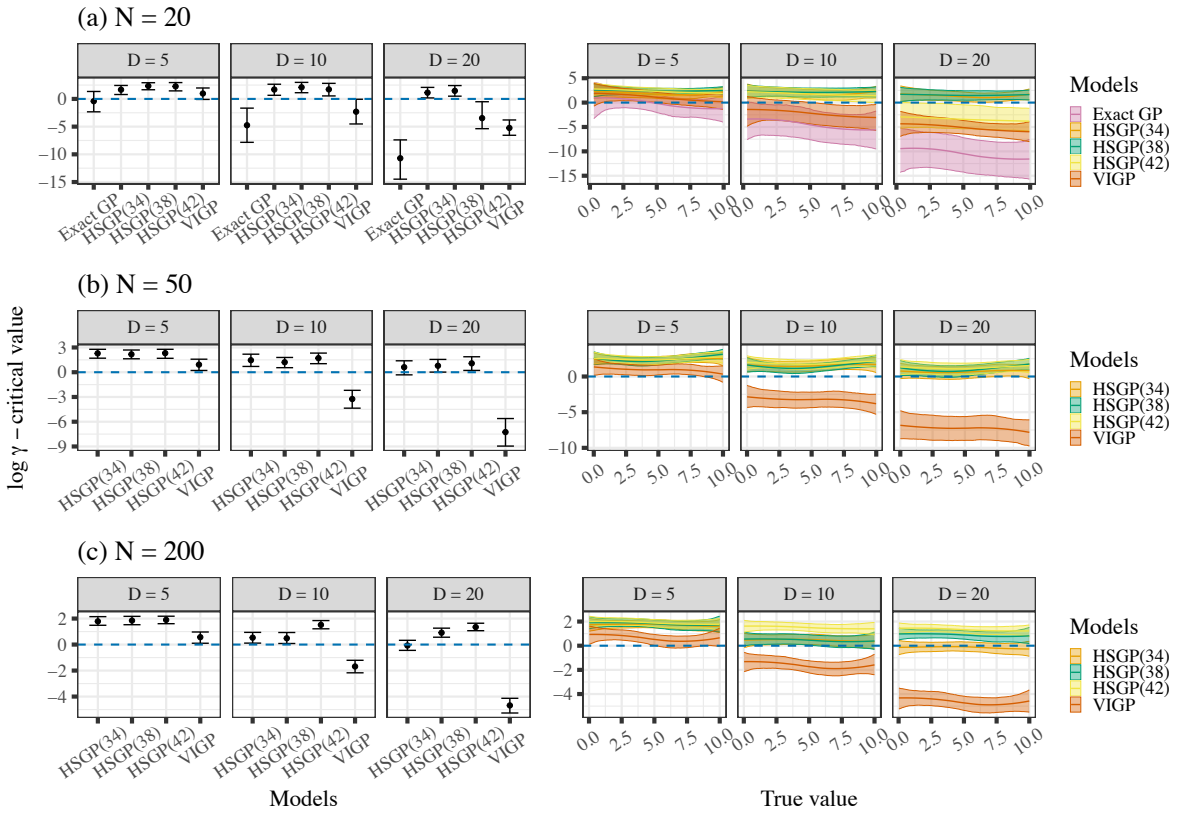

Figure B9: *Matérn 5/2 scenario: log γ scores offset by the 95% confidence threshold for all the fitted models. The behavior of scores across true latent x values are shown in the right-hand panel. The blue dashed line denotes the threshold to reject uniformity. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

B.5: Latent input estimation (additional figures)

We present the model evaluation on latent variable estimation accuracy for the Matérn 3/2, 5/2 as well as the periodic data (both high and low oscillation) scenarios. In almost all of the cases, HSGPs have lower bias compared to exact GPs and VIGPs. In case of posterior SDs, the results are mixed. VIGPs sometimes show lower posterior SD as compared to HSGPs whereas in other cases, they are higher or same. Combined with the calibration test results, VIGPs are likely to underestimate posterior uncertainty due to severe model miscalibrations.

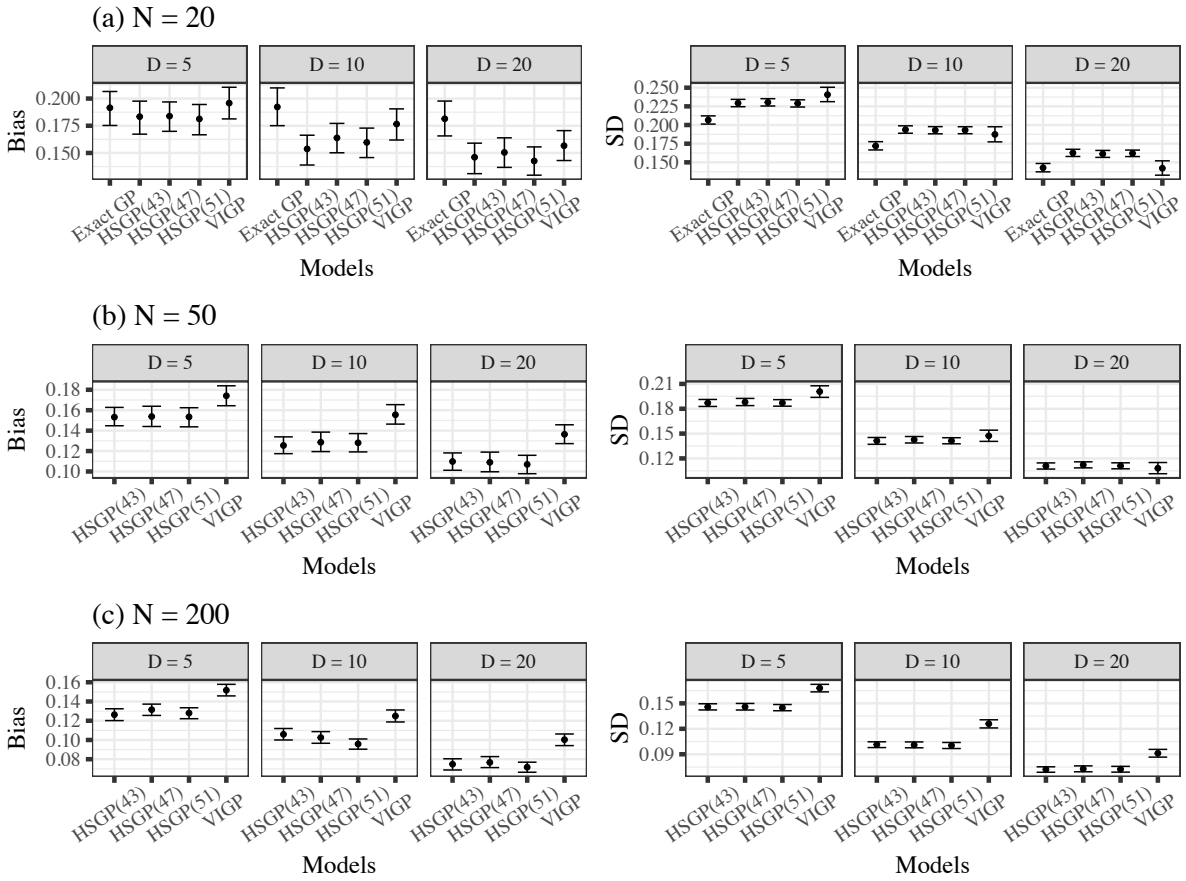

Figure B10: *Matérn 3/2 scenario: posterior bias and SD on recovery of latent inputs for all fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

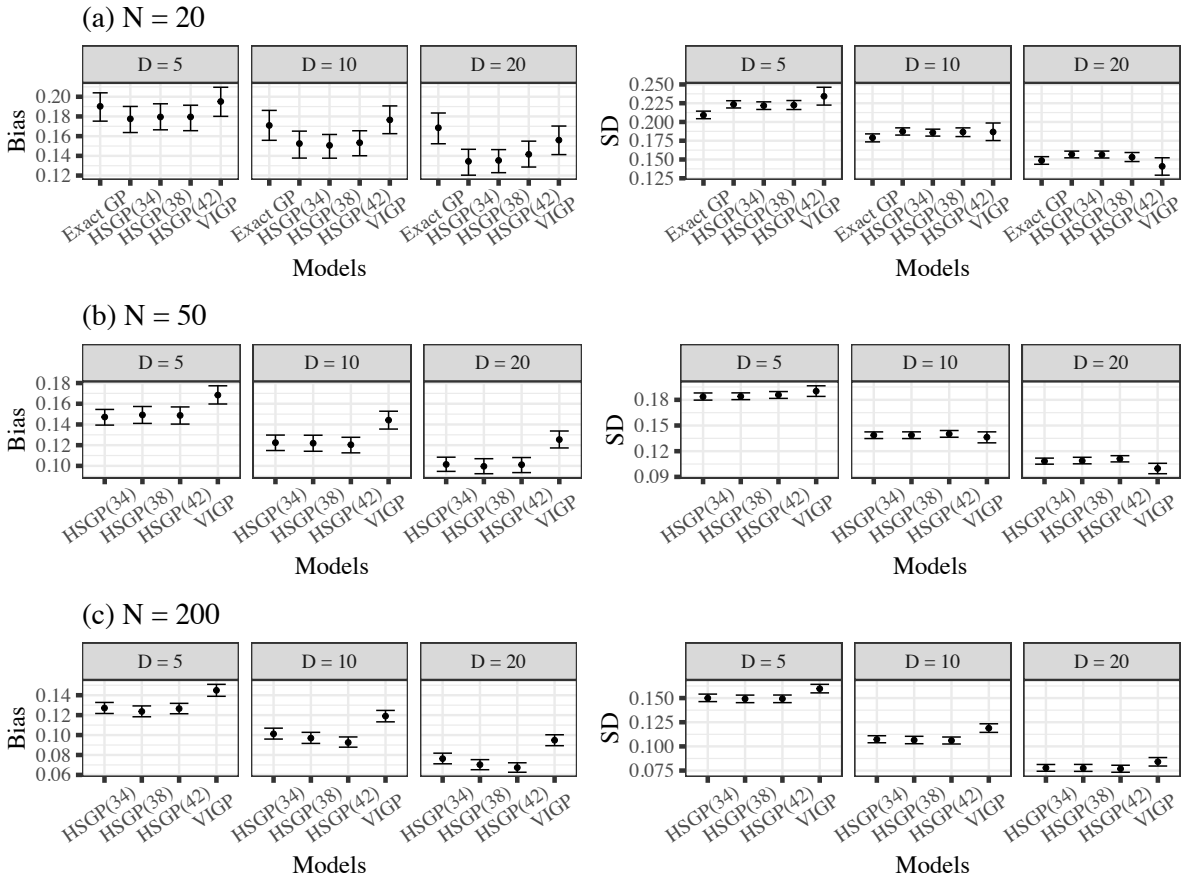

Figure B11: *Matérn 5/2 scenario: posterior bias and SD on recovery of latent inputs for all fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

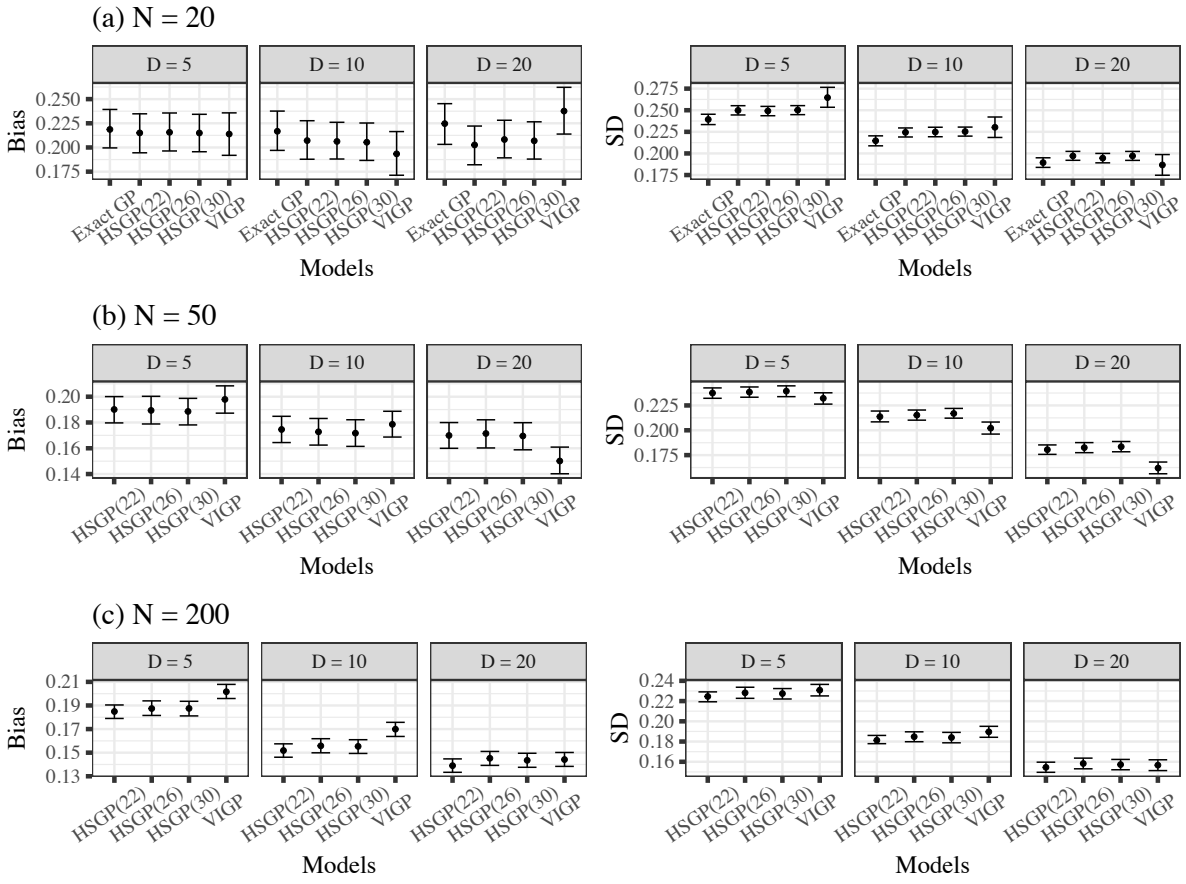

Figure B12: *Periodic data scenario (lower oscillations): posterior bias and SD on recovery of latent inputs for all fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

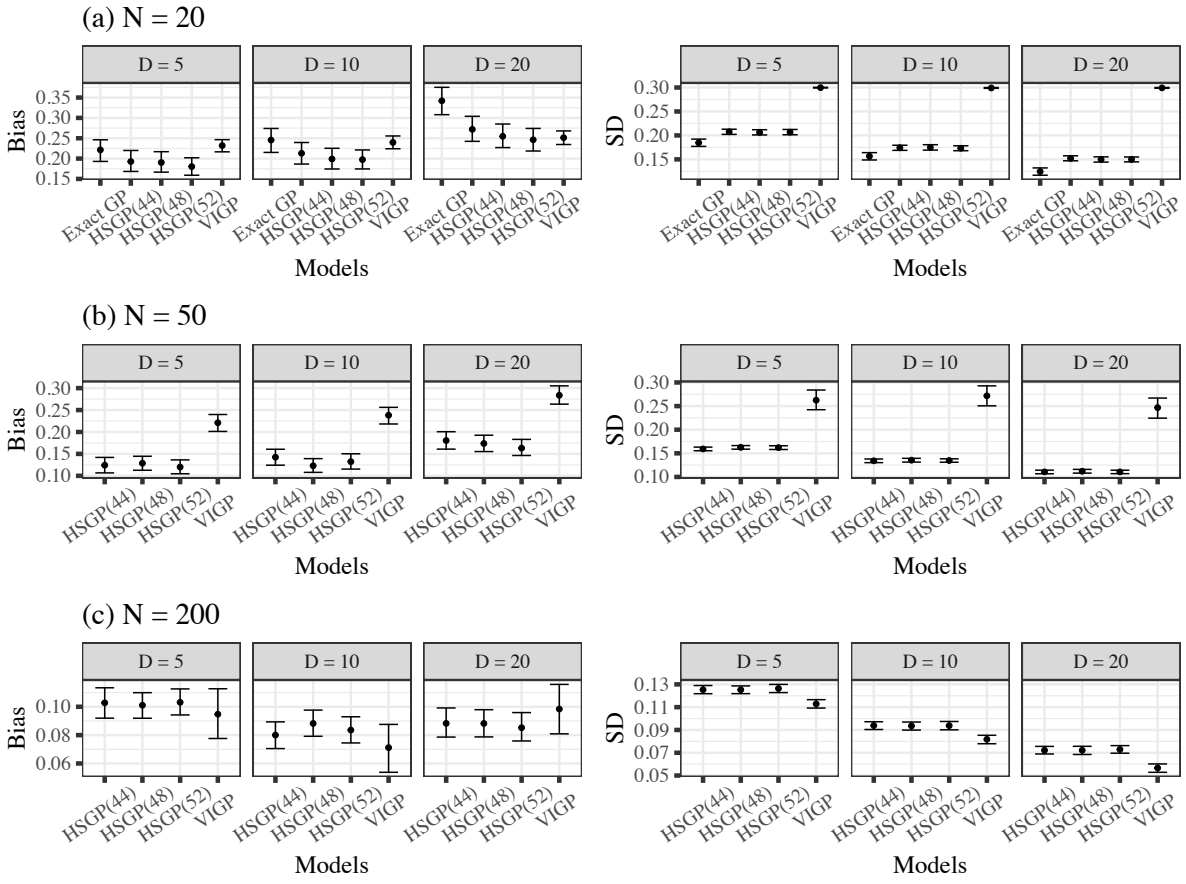

Figure B13: *Periodic data scenario (higher oscillations): posterior bias and SD on recovery of latent inputs for all fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

B.6: Hyperparameter estimation (additional figures)

The GP hyperparameter estimation results for the Matérn 3/2, 5/2 and the periodic data scenarios (both lower and higher oscillations) are presented here. The results are qualitatively similar across all the simulation scenarios.

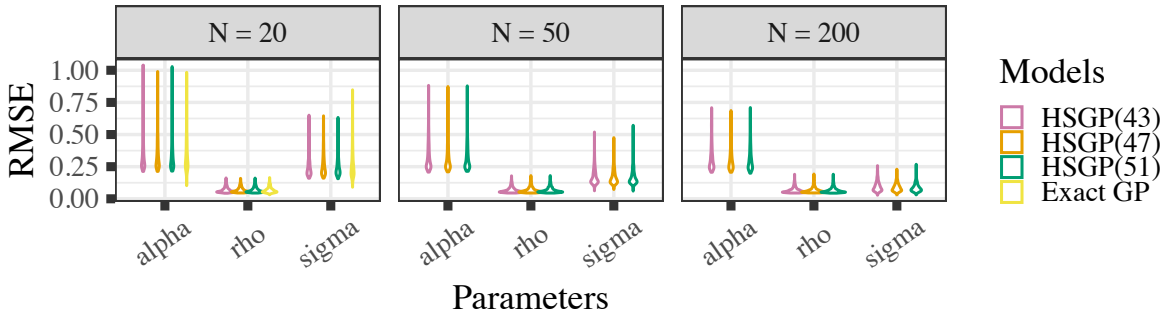

Figure B14: *Matérn 3/2 scenario: RMSE on recovery of GP hyperparameters for exact GP and HSGP fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

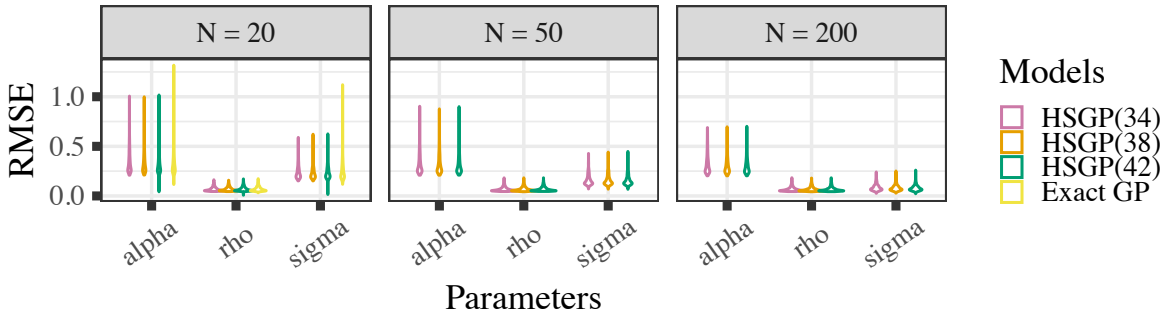

Figure B15: *Matérn 5/2 scenario: RMSE on recovery of GP hyperparameters for exact GP and HSGP fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

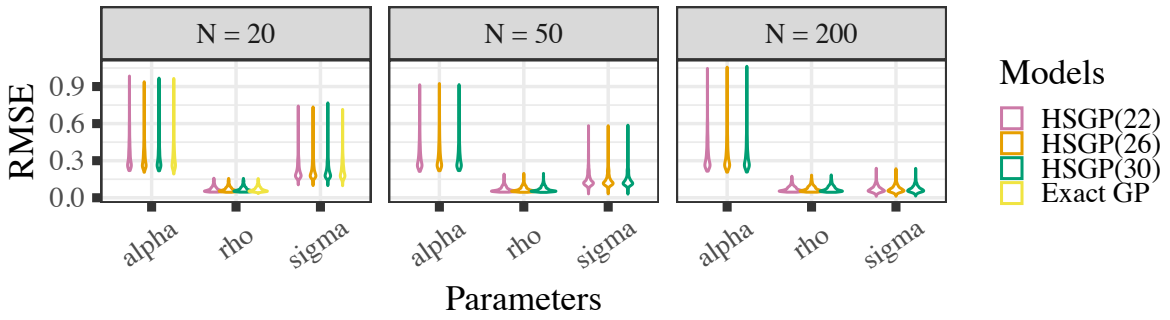

Figure B16: *Periodic data scenario (lower oscillations): RMSE on recovery of GP hyperparameters for exact GP and HSGP fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

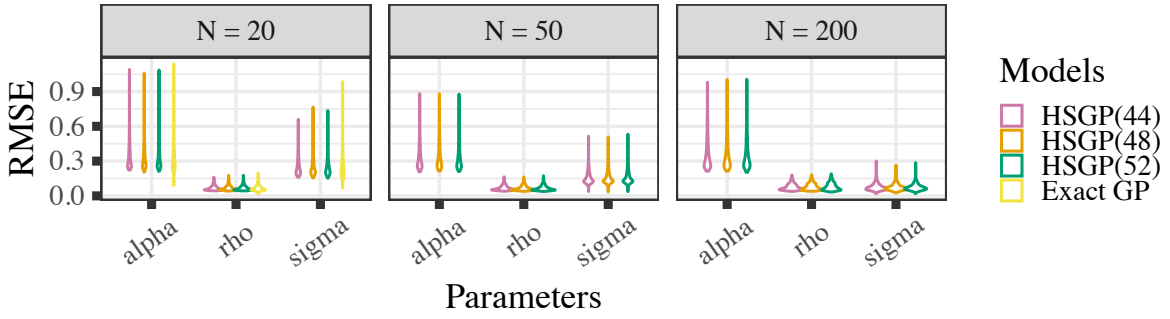

Figure B17: *Periodic data scenario (higher oscillations): RMSE on recovery of GP hyperparameters for exact GP and HSGP fitted models. The HSGP(M) shows the HSGPs with their corresponding number of basis functions.*

B.7: List of genes for the case study

In our real-world case study for cell cycle data, we use the data from all the cells along with a selection of 12 influential genes (see supplementary materials Section G for the list) based on the recommendations of [56]. The list of genes used are: CCNA2, CCNB1, BIRC5, TOP2A, PLK1, NDC80, CDCA8, MKI67, CDC6, CENPF, KRT17, RRM2. These recommendations were primarily concerned with high amount of RNA activities denoted by their counts.

Appendix C

Declaration

This appendix is for Chapter 5 and is based on the supplementary materials of the following preprint that is under review:

> *Latent variable estimation with composite Hilbert space Gaussian processes*
> **Soham Mukherjee**, Javier Enrique Aguilar, Marcello Zago, Manfred Claassen and Paul-Christian Bürkner
> arXiv preprint arXiv:2510.25371 – in review (2025)
> https://doi.org/10.48550/arXiv.2510.25371

Text, figures and tables are adapted from the supplementary materials of the preprint https://arxiv.org/abs/2510.25371.

Supplementary materials

We present the proofs of the theoretical results stated in Chapter 5 in Section C.1. We then discuss model inference strategies in Section C.2 followed by convergence diagnostics related to our simulation studies in Section C.3. The methods used to summarize and present the results from our simulation studies are presented in Section C.4 followed by results related to latent variable uncertainty calibration and covariance function hyperparameter estimation in Sections C.5 and C.6. Finally, in Section C.7, we present additional results related to our real-world case study.

C.1: Derivative kernels and spectral representations

In the following, we show how the spectral representation of stationary covariance kernels can be used to characterize the smoothness of the kernel and to derive explicit expressions for the covariance and spectral densities of derivative Gaussian processes. By using Bochner's theorem we prove regularity conditions ensuring differentiability of the kernel, and derive general forms for the derivative kernels and their spectral representations. We then apply these results to the squared exponential and Matérn classes and conclude by identifying the conditions under which derivative kernels remain positive semidefinite and isotropic.

Let $k(\mathbf{x}, \mathbf{x}') : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ be a continuous, positive-definite covariance function, defining a

mean-zero Gaussian process

$$f(\mathbf{x}) \sim \mathcal{GP}\left(\mathbf{0},\, k(\mathbf{x}, \mathbf{x}')\right).$$

If k is stationary, that is $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}') = k(\mathbf{r})$ with $\mathbf{r} = \mathbf{x} - \mathbf{x}'$, then by Bochner's theorem [31] k is continuous and positive-definite if and only if there exists a finite nonnegative (spectral) measure S on $\mathbb{R}^p$ such that

$$k(\mathbf{r}) = \frac{1}{(2\pi)^p} \int_{\mathbb{R}^p} \exp\left(i\boldsymbol{\omega}^T \mathbf{r}\right)\, S(d\boldsymbol{\omega}).$$

If S is absolutely continuous with respect to the Lebesgue measure, write $S(d\boldsymbol{\omega}) = S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}$, the function $S_k(\boldsymbol{\omega}) \geq 0$ is the spectral density corresponding to the covariance function $k(\mathbf{r})$, and the Fourier pair is [69, 82]

$$\begin{aligned} k(\mathbf{r}) &= \frac{1}{(2\pi)^p} \int_{\mathbb{R}^p} \exp\left(i\boldsymbol{\omega}^T \mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}, \\ S_k(\boldsymbol{\omega}) &= \int_{\mathbb{R}^p} \exp\left(-i\boldsymbol{\omega}^T \mathbf{r}\right) k(\mathbf{r})\, d\mathbf{r}, \qquad \boldsymbol{\omega} \in \mathbb{R}^p. \end{aligned} \tag{6.4}$$

We proceed to define the kernel associated to the derivative of the process $f(\mathbf{x})$. Assume that $f(\mathbf{x})$ is a mean zero GP with stationary kernel $k(\cdot, \cdot)$. Consider the multi-indices $\mathbf{a}, \mathbf{b} \in \mathbb{N}_0^p$ where $\mathbb{N}_0^p = \{(n_1, \ldots, n_p) \mid n_j \in \mathbb{N}_0, j = 1, ..., p\}$. Denote by $g^{(\mathbf{a})}(\mathbf{x}) = \partial^{\mathbf{a}} f(\mathbf{x})$ and $g^{(\mathbf{b})}(\mathbf{x}) = \partial^{\mathbf{b}} f(\mathbf{x})$ where $\partial^{\mathbf{a}}$ represents the following

$$\partial^{\mathbf{a}} f(\mathbf{x}) = \frac{\partial^{|\mathbf{a}|} f(\mathbf{x})}{\partial^{a_1}\mathbf{x}_1 \ldots \partial^{a_p}\mathbf{x}_p}, \tag{6.5}$$

where $|\mathbf{a}| = \sum_{j=1}^p a_j$.

We define the derivative $k^{(\mathbf{a},\mathbf{b})}(\mathbf{x}, \mathbf{x}')$ as

$$k^{(\mathbf{a},\mathbf{b})}(\mathbf{x}, \mathbf{x}') = \operatorname{Cov}\left(g^{\mathbf{a}}(\mathbf{x}), g^{\mathbf{b}}(\mathbf{x}')\right) = \partial_{\mathbf{x}}^{\mathbf{a}}\, \partial_{\mathbf{x}'}^{\mathbf{b}}\, k(\mathbf{x}, \mathbf{x}'). \tag{6.6}$$

We proceed to prove the following lemma.

Lemma 1. *Assume the spectral density is such that $S_k \in L^1(\mathbb{R}^p)$ and that for some integer $m \geq 1$,*

$$\int_{\mathbb{R}^p} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} < \infty.$$

Then, the kernel $k \in C^m(\mathbb{R}^p)$ and, for every multi-index $\mathbf{a}$ with $|\mathbf{a}| \leq m$,

$$\partial_{\mathbf{r}}^{\mathbf{a}} k(\mathbf{r}) = \int_{\mathbb{R}^p} (i\boldsymbol{\omega})^{\mathbf{a}} \exp\left(i\boldsymbol{\omega}^T \mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}, \qquad \mathbf{r} \in \mathbb{R}^p,$$

where $(i\boldsymbol{\omega})^{\mathbf{a}} \coloneqq \prod_{j=1}^p (i\omega_j)^{a_j}$.

Proof. Fix $j \in \{1, \ldots, p\}$ and let $\mathbf{e_j}$ be the jth canonical vector. By the definition of the partial

derivative,

$$\partial_{r_j} k(\mathbf{r}) = \frac{\partial k(\mathbf{r})}{\partial r_j} = \lim_{h\to 0} \frac{k(\mathbf{r}+h\mathbf{e_j}) - k(\mathbf{r})}{h}.$$

Using Bochner's representation theorem we have that

$$\frac{k(\mathbf{r}+h\mathbf{e_j}) - k(\mathbf{r})}{h} = \int_{\mathbb{R}^p} \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) \frac{\exp(ih\boldsymbol{\omega}_j) - 1}{h} S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}.$$

Define the function $Q_h(\mathbf{r},\boldsymbol{\omega}) = \exp(i\boldsymbol{\omega}^T\mathbf{r})\frac{\exp(ih\boldsymbol{\omega}_j)-1}{h} S_k(\boldsymbol{\omega})$. For each fixed $\boldsymbol{\omega}$, $\frac{\exp(ih\omega_j)-1}{h} \to i\omega_j$ as $h \to 0$, therefore $Q_h(\mathbf{r},\boldsymbol{\omega}) \to i\omega_j \exp(i\boldsymbol{\omega}^T\mathbf{r}) S_k(\boldsymbol{\omega})$ pointwise in $\boldsymbol{\omega}$.

Moreover, since $|\exp(i\theta) - 1| = 2\sin(\theta/2)$ and using the fact that $|\sin(\theta)| \le \theta$ we have

$$\left|\frac{\exp(ih\omega_j) - 1)}{h}\right| = \frac{2}{|h|}|\sin(h\omega_j/2)| \le |\omega_j| \quad \text{for all } h \neq 0.$$

This implies that $|Q_h(\mathbf{r},\boldsymbol{\omega})| \le |\omega_j| S_k(\boldsymbol{\omega})$, which is integrable by hypothesis. Applying the dominated convergence theorem shows that

$$\begin{aligned}
\frac{\partial k(\mathbf{r})}{\partial r_j} &= \lim_{h\to 0} \frac{k(\mathbf{r}+h\mathbf{e_j}) - k(\mathbf{r})}{h} \\
&= \lim_{h\to 0} \int Q_h(\mathbf{r},\boldsymbol{\omega}) d\boldsymbol{\omega} \\
&= \int \lim_{h\to 0} Q_h(\mathbf{r},\boldsymbol{\omega}) d\boldsymbol{\omega} \\
&= \int (i\omega_j) \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} \\
&= \int \frac{\partial}{\partial r_j} \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}.
\end{aligned}$$

To prove continuity of $\partial_{r_j} k(\mathbf{r})$, consider a fixed but arbitrary sequence $\{\mathbf{r}_n\}_{n=1}^{\infty}$ such that $\mathbf{r}_n \to \mathbf{r}$. Clearly, $i\omega_j \exp(i\boldsymbol{\omega}^T\mathbf{r}_n) S_k(\boldsymbol{\omega}) \to i\omega_j \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})$ pointwise in $\boldsymbol{\omega}$ and is dominated by the integrable function $|\omega_j| S_k(\boldsymbol{\omega}) \in L^1$. Therefore, we can apply the dominated convergence theorem to show that

$$\begin{aligned}
\lim_{n\to\infty} \partial_{r_j} k(\mathbf{r}_n) &= \lim_{n\to\infty} \int_{\mathbb{R}^p} (i\omega_j) \exp\left(i\boldsymbol{\omega}^T\mathbf{r}_n\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} \\
&= \int_{\mathbb{R}^p} \lim_{n\to\infty} (i\omega_j) \exp\left(i\boldsymbol{\omega}^T\mathbf{r}_n\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} \\
&= \int_{\mathbb{R}^p} (i\omega_j) \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} = \partial_{r_j} k(\mathbf{r}).
\end{aligned}$$

This shows that $\partial_{r_j} k$ is continuous and $k \in C^1(\mathbb{R}^p)$.

Iterating the same argument $|\mathbf{a}|$ times, each step multiplies the integrand by a factor ω_{j_ℓ} and is dominated by $\|\boldsymbol{\omega}\|^{|\mathbf{a}|} S_k(\boldsymbol{\omega}) \in L^1$. Therefore for all $\mathbf{a}$ such that $|\mathbf{a}| \le m$,

$$\partial_{\mathbf{r}}^{\mathbf{a}} k(\mathbf{r}) = \int_{\mathbb{R}^p} (i\boldsymbol{\omega})^{\mathbf{a}} \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega},$$

and $\partial_{\mathbf{r}}^{\mathbf{a}} k$ is continuous. This proves $k \in C^m(\mathbb{R}^p)$ and the stated formula. □

We can now show that the derivative kernel $k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$ exists and how it is defined in terms of $k(\mathbf{r})$.

Proposition 4 (Derivative kernel)**.** *Let $k(\mathbf{x},\mathbf{x}') = k(\mathbf{r})$, where $\mathbf{r} = \mathbf{x} - \mathbf{x}'$, be a stationary covariance kernel on $\mathbb{R}^p$ with spectral density $S_k \in L^1(\mathbb{R}^p)$ such that for some integer $m \geq 1$,*

$$\int_{\mathbb{R}^p} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} < \infty.$$

For multi-indices $\mathbf{a},\mathbf{b} \in \mathbb{N}_0^p$ such that $|\mathbf{a}| + |\mathbf{b}| \leq m$, define

$$k^{(\mathbf{a},\mathbf{b})}(\mathbf{r}) = \mathrm{Cov}\left(g^{\mathbf{a}}(\mathbf{x}), g^{\mathbf{b}}(\mathbf{x}')\right) = \partial_{\mathbf{x}}^{\mathbf{a}} \partial_{\mathbf{x}'}^{\mathbf{b}} k(\mathbf{x},\mathbf{x}').$$

Then

$$k^{(\mathbf{a},\mathbf{b})}(\mathbf{r}) = (-1)^{|\mathbf{b}|} \partial_{\mathbf{r}}^{\mathbf{a}+\mathbf{b}} k(\mathbf{r}),$$

where $|\mathbf{b}| = \sum_{j=1}^p b_j$.

Proof. By Lemma 1, differentiating first with respect to $\mathbf{x}'$ gives

$$\partial_{\mathbf{x}'}^{\mathbf{b}} k(\mathbf{x},\mathbf{x}') = \int_{\mathbb{R}^p} (-i\boldsymbol{\omega})^{\mathbf{b}} \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}.$$

Since $\mathbf{r} = \mathbf{x} - \mathbf{x}'$, each derivative with respect to x_j' introduces a -1 through $\partial_{x_j'} = \partial_{r_j}$. Differentiating again with respect to $\mathbf{x}$ yields

$$\partial_{\mathbf{x}}^{\mathbf{a}} \partial_{\mathbf{x}'}^{\mathbf{b}} k(\mathbf{x},\mathbf{x}') = \int_{\mathbb{R}^p} (i\boldsymbol{\omega})^{\mathbf{a}} (-i\boldsymbol{\omega})^{\mathbf{b}} \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}.$$

Therefore,

$$\partial_{\mathbf{x}}^{\mathbf{a}} \partial_{\mathbf{x}'}^{\mathbf{b}} k(\mathbf{x},\mathbf{x}') = (-1)^{|\mathbf{b}|} \partial_{\mathbf{r}}^{\mathbf{a}+\mathbf{b}} k(\mathbf{r}).$$

□

We now show an explicit formula for the spectral density of $k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$ in terms of the spectral density of $k(\mathbf{r})$.

Proposition 5 (Spectral representation of derivative kernels)**.** *Let $k^{(\mathbf{a},\mathbf{b})}(\mathbf{x},\mathbf{x}') = k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$ be a stationary covariance kernel, then its spectral density $S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega})$ is given by*

$$S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega}) = (i\boldsymbol{\omega})^{\mathbf{a}} (-i\boldsymbol{\omega})^{\mathbf{b}} S_k(\boldsymbol{\omega}).$$

Proof. From Proposition 4,

$$k^{(\mathbf{a},\mathbf{b})}(\mathbf{r}) = \int_{\mathbb{R}^p} (i\boldsymbol{\omega})^{\mathbf{a}} (-i\boldsymbol{\omega})^{\mathbf{b}} \exp\left(i\boldsymbol{\omega}^T\mathbf{r}\right) S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}.$$

By the uniqueness statement in Bochner's theorem, this implies that

$$S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega}) = (i\boldsymbol{\omega})^{\mathbf{a}} (-i\boldsymbol{\omega})^{\mathbf{b}} S_k(\boldsymbol{\omega}).$$

□

We now show when Lemma 1 holds for the SE and Matérn kernels.

Corollary 1 (Squared Exponential)**.** *Let $k(\mathbf{r})$ represent the SE kernel given by,*

$$k(\mathbf{r}) = \alpha^2 \exp\left(-\frac{\|\mathbf{r}\|^2}{2\rho^2}\right), \qquad \mathbf{r} \in \mathbb{R}^p, \tag{6.7}$$

then Lemma 1 holds for every integer $m \geq 0$. In particular $k \in C^\infty(\mathbb{R}^p)$ and, for all multi-indices $\mathbf{a}, \mathbf{b} \in \mathbb{N}_0^p$,

$$S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega}) = (i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}} S_k(\boldsymbol{\omega}).$$

Proof. The spectral density of $k(\mathbf{r})$ is the Fourier transform of an unnormalized Gaussian distribution, given by

$$\begin{aligned} S_k(\boldsymbol{\omega}) &= \int_{\mathbb{R}^p} \exp\left(-i\boldsymbol{\omega}^T\mathbf{r}\right) k(\mathbf{r})\, d\mathbf{r} \\ &= \int_{\mathbb{R}^p} \exp(-i\boldsymbol{\omega}^T\mathbf{r})\alpha^2 \exp\left(-\frac{1}{2\rho^2}\|\mathbf{r}\|^2\right) d\mathbf{r} \\ &= (2\pi)^{p/2}\alpha^2\rho^p \exp\left(-\frac{1}{2}\rho^2\|\boldsymbol{\omega}\|^2\right). \end{aligned}$$

Being Gaussian, it has finite polynomial moments of all orders [81], hence

$$\int_{\mathbb{R}^p} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} < \infty \quad \text{for every } m \geq 0.$$

Therefore the hypothesis of Lemma 1 is satisfied for any m, yielding $k \in C^\infty(\mathbb{R}^d)$ and the stated identity. □

Corollary 2 (Matérn)**.** *Let $k(\mathbf{r})$ be the Matérn kernel on $\mathbb{R}^p$ with variance α^2, smoothness $\nu > 0$ and scale $\rho > 0$. Then Lemma 1 holds for every integer m with $0 \leq m < 2\nu$. For all multi-indices $\mathbf{a}, \mathbf{b} \in \mathbb{N}_0^p$ with $|\mathbf{a}| + |\mathbf{b}| < 2\nu$,*

$$S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega}) = (i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}}\, S_k(\boldsymbol{\omega})$$

and $\partial_{\mathbf{r}}^{\mathbf{a}} k, \partial_{\mathbf{r}}^{\mathbf{b}} k$ exist and are continuous. In particular, $k \in C^m(\mathbb{R}^p)$ for any $m < 2\nu$.

We first establish the following auxiliary lemma whose proof appears in [81].

Lemma 2. *Let $f : \mathbb{R}^p \to \mathbb{R}$ be an integrable function that depends only on the Euclidean norm, i.e. $f(\boldsymbol{\omega}) = f(\|\boldsymbol{\omega}\|)$. Then*

$$\int_{\mathbb{R}^p} f\left(\|\boldsymbol{\omega}\|\right) d\boldsymbol{\omega} = \vartheta_{p-1} \int_0^\infty f(l)\ell^{p-1} d\ell$$

where ϑ_{p-1} denotes the surface area of the $(p-1)$–dimensional unit sphere,

$$\vartheta_{p-1} = \frac{2\pi^{p/2}}{\Gamma(p/2)}.$$

In particular if $B_\varepsilon(0)$ denotes the ball of radius $\varepsilon > 0$ centered at the origin we have

$$\int_{B_\varepsilon(0)} f(\boldsymbol{\omega})\, d\boldsymbol{\omega} = \vartheta_{p-1} \int_0^\varepsilon f(\ell)\ell^{p-1} d\ell.$$

Proof. The Matérn class of covariance functions is given by

$$k(\mathbf{r}) = \alpha^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{\rho} \|\mathbf{r}\| \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}}{\rho} \|\mathbf{r}\| \right),$$

where K is a modified Bessel function [1]. Its spectral density is given by [38]

$$S_k(\boldsymbol{\omega}) = \alpha^2 \frac{\Gamma\left(\nu + \frac{p}{2}\right)(2\nu)^\nu \rho^{-2\nu}}{\Gamma(\nu)\,\pi^{p/2}} \left(\frac{2\nu}{\rho^2} + \|\boldsymbol{\omega}\|^2 \right)^{-(\nu+p/2)}.$$

Let $I_m = \int_{\mathbb{R}^p} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega}$. We split the domain of integration of I_m into the unit ball $B_1(0) = \{\boldsymbol{\omega} \mid \|\boldsymbol{\omega}\| \leq 1\}$ and its complement:

$$\begin{aligned} I_m &= \int_{\mathbb{R}^p} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} \\ &= \int_{\|\boldsymbol{\omega}\|\leq 1} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} + \int_{\|\boldsymbol{\omega}\|>1} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} \end{aligned}$$

We first consider the integral over $B_1(0)$. Since $S_k(\boldsymbol{\omega})$ is continuous on the compact set $B_1(0)$, it is bounded by the Extreme Value theorem. Let $M_0 = \sup_{\|\boldsymbol{\omega}\|\leq 1} S_k(\boldsymbol{\omega}) < \infty$. Applying Lemma 2 we obtain

$$\begin{aligned} \int_{\|\boldsymbol{\omega}\|\leq 1} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} &\leq \int_{\|\boldsymbol{\omega}\|\leq 1} \|\boldsymbol{\omega}\|^m M_0\, d\boldsymbol{\omega} \\ &\leq M_0 \vartheta_{p-1} \int_0^1 \ell^{m+p-1} d\ell \\ &\leq M_0 \frac{2\pi^{p/2}}{\Gamma(p/2)} \frac{1}{m+p} < \infty. \end{aligned}$$

We now consider the integral on the complement of $B_1(0)$. We can rewrite the following term involved in $S_k(\boldsymbol{\omega})$ as $\frac{2\nu}{\rho^2} + \|\boldsymbol{\omega}\|^2 = \|\boldsymbol{\omega}\|^2 \left(\frac{2\nu}{\rho^2} \frac{1}{\|\boldsymbol{\omega}\|^2} + 1 \right)$.

Since $\|\boldsymbol{\omega}\| > 1$ implies $\frac{2\nu}{\rho^2} \frac{1}{\|\boldsymbol{\omega}\|^2} + 1 \geq 1$, we have

$$\left(\frac{2\nu}{\rho^2} + \|\boldsymbol{\omega}\|^2 \right)^{-(\nu+p/2)} = \left(\|\boldsymbol{\omega}\|^2 \left(\frac{2\nu}{\rho^2} \frac{1}{\|\boldsymbol{\omega}\|^2} + 1 \right) \right)^{-(\nu+p/2)} \leq \|\boldsymbol{\omega}\|^{-2\nu-p}.$$

This provides the following tail bound when $\|\boldsymbol{\omega}\| > 1$

$$S_k(\boldsymbol{\omega}) = C \left(\frac{2\nu}{\rho^2} + \|\boldsymbol{\omega}\|^2 \right)^{-(\nu+p/2)} \leq C \|\boldsymbol{\omega}\|^{-2\nu-p},$$

where $C = \alpha^2 \frac{\Gamma\left(\nu+\frac{p}{2}\right)(2\nu)^\nu \rho^{-2\nu}}{\Gamma(\nu)\pi^{p/2}}$. Combining this bound and applying Lemma 2 gives

$$\begin{aligned}\int_{\|\boldsymbol{\omega}\|>1} \|\boldsymbol{\omega}\|^m S_k(\boldsymbol{\omega})\, d\boldsymbol{\omega} &\leq \int_{\|\boldsymbol{\omega}\|>1} C\|\boldsymbol{\omega}\|^{m-2\nu-p}\, d\boldsymbol{\omega} \\ &\leq C\vartheta_{p-1}\int_1^\infty \ell^{m-2\nu-1}\, d\ell \\ &\leq C\frac{2\pi^{p/2}}{\Gamma(p/2)}\int_1^\infty \ell^{m-2\nu-1}\, d\ell.\end{aligned}$$

The last integral converges if and only if $m < 2\nu$.

Combining the two regions shows $I_m < \infty$ when $m < 2\nu$, which verifies the moment condition in Lemma 1.

□

Even though we can construct $k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$, it does not imply that it represents a covariance kernel. The following proposition provides sufficient and necessary conditions for this to occur.

Proposition 6 (Conditions for $k^{(\mathbf{a},\mathbf{b})}$ to be a covariance kernel)**.** *Assume the kernel k is stationary and isotropic with spectral density $S_k(\boldsymbol{\omega})$ and that $|\mathbf{a}| + |\mathbf{b}| \leq m$. Then the derivative function $k^{(\mathbf{a},\mathbf{b})}$ is a positive semidefinite covariance kernel if and only if the following set of conditions hold:*

(1) $a_j + b_j$ *is even for every* $j = 1, \ldots, p$.

(2) The multi-indices $\mathbf{a}, \mathbf{b}$ *are such that* $|\mathbf{a}| + 3|\mathbf{b}| \equiv 0 \pmod 4$.

In particular, $\mathbf{a} = \mathbf{b}$ *always satisfies these conditions, so* $k_{\mathbf{a},\mathbf{a}}$ *is a covariance kernel.*

Proof. From Proposition 5 the candidate for spectral density of $k^{(\mathbf{a},\mathbf{b})}$ is

$$S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega}) = (i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}} S_k(\boldsymbol{\omega}).$$

To show the necessity and sufficiency of the conditions we will make use of the fact that a stationary kernel is positive semidefinite if and only if its spectral density is real and nonnegative almost everywhere. Instead of working directly with $k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$, we will work with $S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega})$.

We will first show the necessity of the condition (1). We can write the multiplier $(i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}}$ coordinatewise as:

$$(i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}} = \prod_{j=1}^{p} i^{a_j}(-i)^{b_j}\omega_j^{a_j+b_j}.$$

Let $R_j\boldsymbol{\omega}$ be the operator that shifts the sign of the jth element of $\boldsymbol{\omega}$:

$$R_j\boldsymbol{\omega} = (\omega_1, ..., -\omega_j, ..., \omega_p)'.$$

Assume that there exists an index j such that $a_j + b_j$ is odd, then the multiplier associated to

$R_j\boldsymbol{\omega}$ is such that:

$$\begin{aligned}(iR_j\boldsymbol{\omega})^{\mathbf{a}}(-iR_j\boldsymbol{\omega})^{\mathbf{b}} &= \left(\prod_{l\neq j}^{p}(i\omega_l)^{a_l}(-i\omega_l)^{b_l}\right)(-i\omega_j)^{a_j}(i\omega_j)^{b_j}\\ &= (-1)^{a_j+b_j}\left(\prod_{l\neq j}^{p}(i\omega_l)^{a_l}(-i\omega_l)^{b_l}\right)(i\omega_j)^{a_j}(-i\omega_j)^{b_j}\\ &= -\prod_{l=1}^{p}(i\omega_l)^{a_l}(-i\omega_l)^{b_l}\\ &= -(i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}}.\end{aligned}$$

Since $R_j\boldsymbol{\omega}$ and $\boldsymbol{\omega}$ have the same norm, the hypothesis that $k(\mathbf{r})$ is isotropic implies that

$$S_k(R_j\boldsymbol{\omega}) = S_k(\boldsymbol{\omega}).$$

Therefore,

$$S^{(\mathbf{a},\mathbf{b})}(R_j\boldsymbol{\omega}) = (iR_j\boldsymbol{\omega})^{\mathbf{a}}(-iR_j\boldsymbol{\omega})^{\mathbf{b}}S_k(R_j\boldsymbol{\omega}) = -(i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}}S_k(\boldsymbol{\omega}) = -S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega}).$$

This implies that $S^{(\mathbf{a},\mathbf{b})}(\boldsymbol{\omega})$ can not be nonnegative almost everywhere. This shows it is necessary that $a_j + b_j$ is even for all $j = 1, ..., p$.

We proceed to show the sufficiency of condition(1). Assume that a_j+b_j is even for all $j = 1, ...p$. We can rewrite the factor as

$$\begin{aligned}(i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}} &= \prod_{j=1}^{p} i^{a_j}(-i)^{b_j}\omega_j^{a_j+b_j} = \prod_{j=1}^{p}(i)^{a_j}(-i)^{b_j}\prod_{j=1}^{p}|\omega_j|^{a_j+b_j}\\ &= i^{|a|}(-i)^{|b|}\prod_{j=1}^{p}|\omega_j|^{a_j+b_j} = C\prod_{j=1}^{p}|\omega_j|^{a_j+b_j},\end{aligned}$$

where $C = (-1)^{|\mathbf{b}|}i^{|\mathbf{a}|+|\mathbf{b}|}$. The multiplier will be complex valued if and only if $i^{|\mathbf{a}|+|\mathbf{b}|}$ is also complex valued, but the only possible values it can attain are $\{-1, 1\}$. This shows that condition (1) is sufficient for $k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$ to be real valued.

We now show that condition (2) is sufficient and necessary for nonnegativity of $k^{(\mathbf{a},\mathbf{b})}(\mathbf{r})$.

We can rewrite $C = (-1)^{|\mathbf{b}|}i^{|\mathbf{a}|+|\mathbf{b}|} = (-1)^{|\mathbf{b}|}(-1)^{1/2(|\mathbf{a}|+|\mathbf{b}|)} = (-1)^{|\mathbf{b}|+1/2(|\mathbf{a}|+|\mathbf{b}|)}$. Therefore $C = 1$ if and only if $|\mathbf{b}| + 1/2(|\mathbf{a}| + |\mathbf{b}|)$ is even, which is equivalent to

$$|\mathbf{b}| + 1/2(|\mathbf{a}| + |\mathbf{b}|) \equiv 0 \pmod 2 \quad \Longleftrightarrow \quad |\mathbf{a}| + 3|\mathbf{b}| \equiv 0 \pmod 4,$$

which is condition (2). This shows conditions (1) and (2) are necessary and sufficient.

Finally, if $\mathbf{a} = \mathbf{b}$, then $a_j + b_j = 2a_j$ is even for all j and $|\mathbf{a}| + 3|\mathbf{a}| = 4|\mathbf{a}| \equiv 0 \pmod 4$, so $k^{(\mathbf{a},\mathbf{a})}$ is positive semidefinite.

□

If $k^{(\mathbf{a},\mathbf{b})}$ is a covariance kernel, then it will be stationary since multiplication by $(i\boldsymbol{\omega})^{\mathbf{a}}(-i\boldsymbol{\omega})^{\mathbf{b}}$ does not introduce any dependence on $\mathbf{x}$ or $\mathbf{x}'$. The function $k^{(\mathbf{a},\mathbf{b})}$ might not be isotropic, even if $k(\mathbf{r})$ is. To see this assume that k is isotropic with $k(\mathbf{r}) = \varphi(\|\mathbf{r}\|^2)$, with $\varphi(\cdot)$ a sufficiently smooth function and let $p = 1, a = 1, b = 0$, then

$$k^{(1,0)}(\mathbf{x}, \mathbf{x}') = \frac{\partial k(\mathbf{r})}{\partial \mathbf{x}} = \frac{d}{dr}\varphi(\mathbf{r}^2) = 2\mathbf{r}\varphi'(\mathbf{r}^2).$$

Since $k^{(1,0)}$ is odd it can not be isotropic.

The main text considers $k^{(1,1)}$ with $p = 1$. When $k(\mathbf{r})$ is isotropic, then $k^{(1,1)}$ is isotropic as well, since

$$k^{(1,1)}(\mathbf{x}, \mathbf{x}') = \frac{\partial^2 \varphi(\mathbf{r}^2)}{\partial \mathbf{x} \partial \mathbf{x}'} = \frac{\partial \varphi(\mathbf{r}^2)}{\partial \mathbf{r}^2} = -2\varphi'(\mathbf{r}^2) - 4\mathbf{r}^2\varphi''(\mathbf{r}^2). \tag{6.8}$$

The Hilbert space approximation methods proposed by [78] rely on the assumptions that the covariance function $k(\cdot, \cdot)$ is stationary and isotropic, and that its spectral density can be expressed as a radial function of the form $S_k(\|\boldsymbol{\omega}\|) = \psi(\|\boldsymbol{\omega}\|^2)$. If $\psi(\|\boldsymbol{\omega}\|^2)$ admits a polynomial expansion,

$$\psi(\|\boldsymbol{\omega}\|^2) = \sum_{i=0}^{\infty} a_i \left(\|\boldsymbol{\omega}\|^2\right)^i,$$

which holds, for instance, when ψ is analytic, then the corresponding spectral density can be written as

$$S_k(\|\boldsymbol{\omega}\|) = \sum_{i=0}^{\infty} a_i \left(\|\boldsymbol{\omega}\|^2\right)^i.$$

Since $S_{k^{(1,1)}}(\boldsymbol{\omega}) = \omega^2 S_k(\boldsymbol{\omega})$, it follows that $S_{k^{(1,1)}}(\boldsymbol{\omega})$ also admits a polynomial expansion of the same form. This implies that the Hilbert space approximation framework introduced by [78] is equally valid for derivative covariance functions such as $k^{(1,1)}$.

For any isotropic covariance function, the spectral density depends only on the frequency norm $\|\boldsymbol{\omega}\|$, so that a function $\psi(t) = S_k(\sqrt{t})$ with $t = \|\boldsymbol{\omega}\|^2$ can always be defined. The additional assumption that ψ admits a polynomial expansion requires ψ to be analytic in a neighborhood of $t = 0$, which in turn implies that $S_k(\|\boldsymbol{\omega}\|)$ is infinitely differentiable around the origin. This condition is satisfied by the squared exponential covariance, whose spectral density is Gaussian and hence analytic [81].

Isotropy does not hold in general for $p > 1$. In particular, if $a = \mathbf{e_j}, b = \mathbf{e_l}$ it can be shown that

$$k^{(\mathbf{e_j},\mathbf{e_l})}(\mathbf{r}^2) = \frac{\partial^2 k(\mathbf{x}, \mathbf{x}')}{\partial x_j \partial x_l'} = -\frac{\partial \varphi(\|\mathbf{r}\|^2)}{\partial r_j r_l} = -2\delta_{jl}\varphi'(\|\mathbf{r}\|^2) - 4r_j r_l \varphi''(\|\mathbf{r}\|^2).$$

The term $r_j r_\ell$ introduces directional dependence, implying that $k_{j\ell}$ is not invariant under rotations of $\mathbf{r}$. Therefore, even if k is isotropic, the mixed derivative kernels $k_{j\ell}$ with $j \neq \ell$ are not

isotropic.

We can however, form isotropic kernels from non isotropic ones. An example of this is the following one, which is closely related to the usual Laplacian operator [28]:

$$\begin{aligned}\Lambda_{(\mathbf{x},\mathbf{x}')}\, k(\mathbf{x},\mathbf{x}') &= \sum_{j=1}^{p} k^{(\mathbf{e_j},\mathbf{e_j})} \\ &= \sum_{j=1}^{p} -2\varphi'(\|\mathbf{r}\|^2) - 4r_j^2\varphi''(\|\mathbf{r}\|^2). \\ &= -2p\varphi'(\|\mathbf{r}\|^2) - 4\|\mathbf{r}\|^2\varphi''(\|\mathbf{r}\|^2),\end{aligned}$$

which depends only on $\|\mathbf{r}\|^2$ and is therefore isotropic. It is also possible to construct isotropic derivative kernels from linear combinations of powers of the Laplacian operator,

$$P(\Delta) = \sum_{j=0}^{m} a_j\, \Delta^j,$$

where Δ denotes the Laplacian and $a_j \in \mathbb{R}$ are constants. Such operators are invariant to rotations and therefore preserve isotropy [30].

C.2: Inference

In our proposed methods, the primary parameters of interest are the latent inputs $\mathbf{x}$ and GP hyperparameters $\boldsymbol{\theta}$ given data (outputs) $\mathbf{y}_f$ and $\mathbf{y}_g$ and observed noisy inputs $\tilde{\mathbf{x}}$. Our set of hyperparameters include multi-output GP length scales $\boldsymbol{\rho}_f$ and $\boldsymbol{\rho}_g$, GP marginal variances $\boldsymbol{\alpha}_f$ and $\boldsymbol{\alpha}_g$ as well as error variances $\boldsymbol{\sigma}_f$ and $\boldsymbol{\sigma}_g$ corresponding to the HSGPs $\mathbf{f}$ and $\mathbf{g}$ respectively. For the d^{th} output dimension, we use independent model hyperparameter priors

$$\boldsymbol{\theta}_d \sim p(\boldsymbol{\theta_d}) = p(\rho_{f_d})\, p(\alpha_{f_d})\, p(\sigma_{f_d})\, p(\rho_{g_d})\, p(\alpha_{g_d})\, p(\sigma_{g_d}). \tag{6.9}$$

The joint probability density factorizes as

$$p(\mathbf{y}_f, \mathbf{y}_g, \mathbf{x}, \boldsymbol{\theta} \mid \tilde{\mathbf{x}}) = p(\mathbf{x} \mid \tilde{\mathbf{x}}) \prod_{d}^{D} p(\mathbf{y}_{f_d} \mid \mathbf{x}, \boldsymbol{\theta}_d)\, p(\mathbf{y}_{g_d} \mid \mathbf{x}, \boldsymbol{\theta}_d)\, p(\boldsymbol{\theta}_d), \tag{6.10}$$

where $p(\mathbf{y}_{f_d} \mid \mathbf{x}, \boldsymbol{\theta}_d)$ and $p(\mathbf{y}_{f_d} \mid \mathbf{x}, \boldsymbol{\theta}_d)$ denotes the GP-based likelihoods for a single output dimension and $p(\mathbf{x} \mid \tilde{\mathbf{x}})$ denotes the prior for the latent $\mathbf{x}$ implied by the measurement model in Eq.(5.21) of in Chapter 5.

The independent structure in the joint probability density occurs due to our relaxed specification of the composite GPs $\mathbf{f}$ and $\mathbf{g}$. The details of prior specifications used in our experiments are further discussed in Section 5.4 of Chapter 5. Using Bayes' theorem, we obtain the joint posterior over $\mathbf{x}$ and $\boldsymbol{\theta}$ as

$$p(\mathbf{x}, \boldsymbol{\theta} \mid \mathbf{y}, \tilde{\mathbf{x}}) = \frac{p(\mathbf{y}_f, \mathbf{y}_g, \mathbf{x}, \boldsymbol{\theta} \mid \tilde{\mathbf{x}})}{\int\int p(\mathbf{y}_f, \mathbf{y}_g, \mathbf{x}, \boldsymbol{\theta} \mid \tilde{\mathbf{x}})\, d\mathbf{x}\, d\boldsymbol{\theta}}. \tag{6.11}$$

For the special case of derivative HSGPs, we replace the marginal and error variances with

$\boldsymbol{\alpha}^{(1)}$ and $\boldsymbol{\sigma}^{(1)}$ respectively corresponding to the derivative process $\mathbf{f}^{(1)}$. A key difference for the derivative HSGP is that it also shares the same length scale $\boldsymbol{\rho}$ between both $\mathbf{f}$ and $\mathbf{f}^{(1)}$. This occurs due to the fundamental derivative covariance structure for $\mathbf{f}^{(1)}$ which acts as an extension of the chosen base covariance structure for $\mathbf{f}$. Thus, we modify Eq.(6.9) for the hyperparameters of derivative HSGPs as

$$p(\boldsymbol{\theta}_d) = p(\rho_d)\, p(\alpha_d)\, p(\sigma_d)\, p(\alpha_d^{(1)})\, p(\sigma_d^{(1)}). \tag{6.12}$$

The rest of the inference procedure remains similar to Eq.(6.10) and Eq.(6.11) by replacing $\mathbf{y}_g$ with $\mathbf{y}^{(1)}$ and modifying $\boldsymbol{\theta}$ with derivative GP hyperparameters. Posterior samples of $\mathbf{x}$ and $\boldsymbol{\theta}$ for all output dimensions are obtained via MCMC sampling, specifically the adaptive Hamiltonian Monte Carlo (HMC) sampler [14, 42, 62] implemented in the probabilistic programming language Stan [22, 80].

C.3: Model convergence

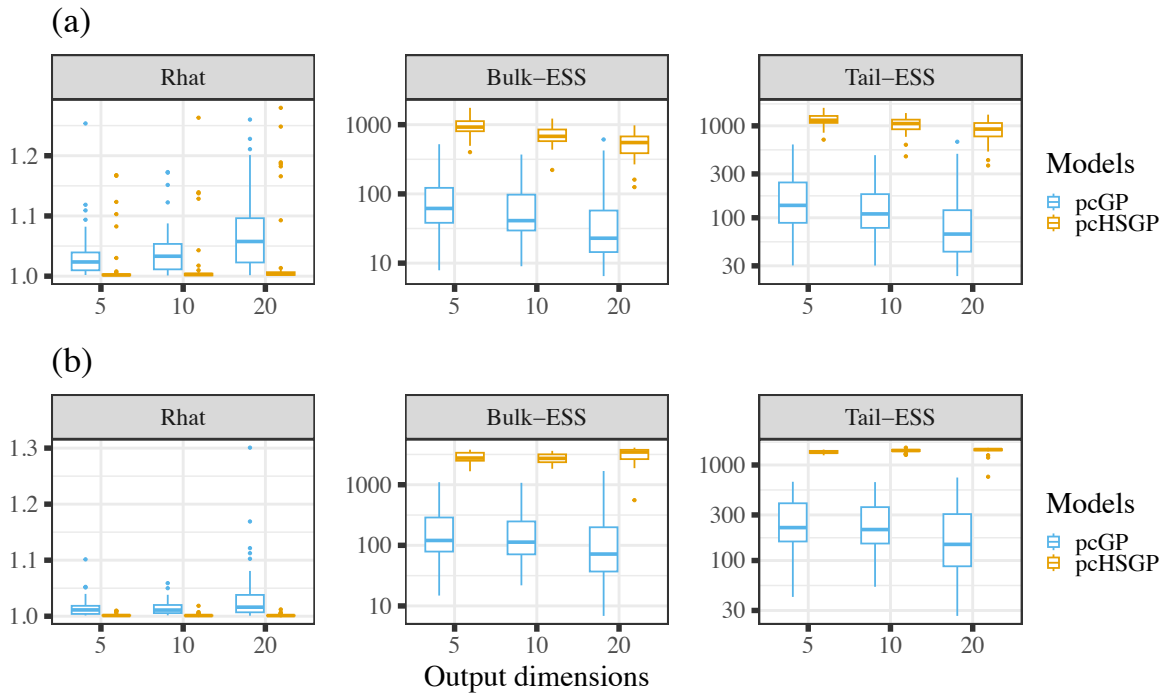

Figure C1: *pcGP data scenario: Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

We investigate MCMC convergence of our fitted exact GPs and HSGPs for all of the simulation study scenarios discussed before. We use standard MCMC sampling diagnostics including latest versions of the scale reduction factor $\widehat{R}$, bulk effective sample size (Bulk-ESS) and tail effective sample size (Tail-ESS) [91]. A combined check of these measures provide a comprehensive summary of the parameter-specific model convergence.

In general, $\widehat{R}$ should be very close to 1 and should ideally not exceed 1.01 [91]. Since exact latent GPs have complex posteriors [60, 61], we additionally consider a more relaxed threshold of 1.1 in our simulation studies. Bulk-ESS indicates the reliability of measures of central tendency such

as the posterior mean or median. Tail-ESS indicates the reliability of the 5% and 95% quantile estimates, which are then used to construct credible intervals. Both Bulk-ESS and Tail-ESS should have values greater than 100 times the number of MCMC chains (higher is better). All of the convergence metrics are considered in combination with comparison to ground truth as another layer of evaluation.

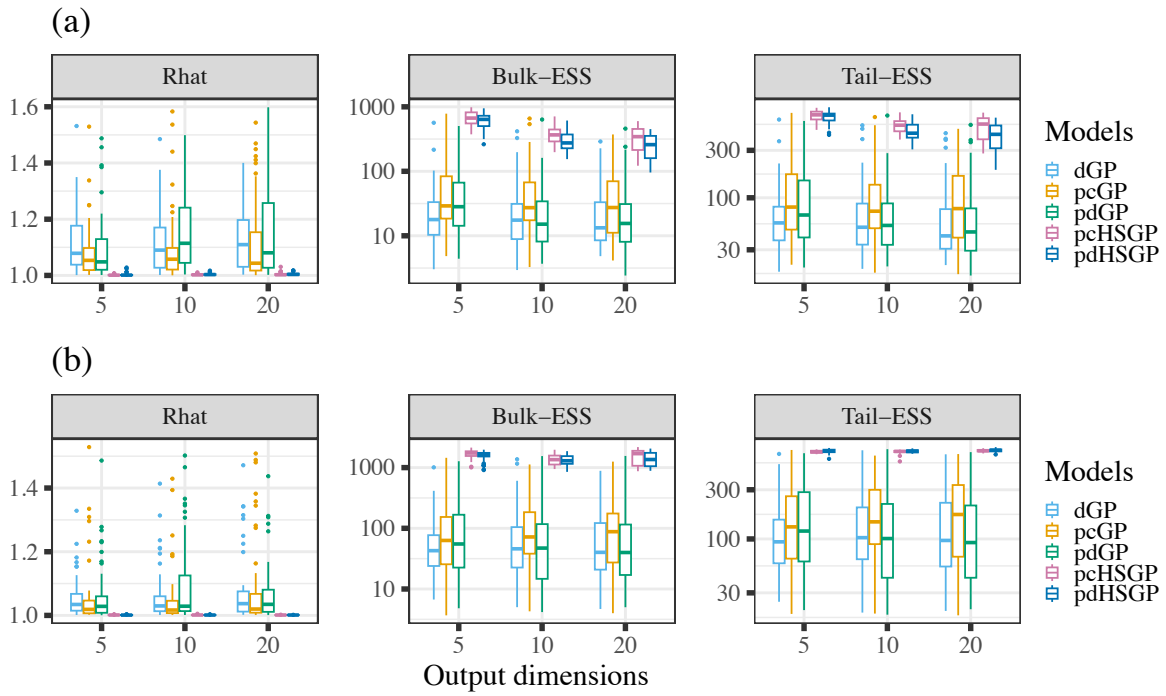

Figure C2: *dGP data scenario: Convergence check for (a) latent inputs and (b) GP hyperparameters of the exact GPs and HSGPs. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

We show $\widehat{R}$, Bulk-ESS, and Tail-ESS for the latent $\mathbf{x}$ and GP hyperparameters for the partial composite GP (pcGP) data (in Fig.C1) and full derivative GP (dGP) data (in Figures C2-C3). While the exact GPs (including the true data generating processes) reach and exceed the relaxed $\widehat{R}$ threshold of 1.1 for some simulated datasets, the approximate versions pcHSGP and pdHSGP consistently satisfy the much stricter 1.01 threshold of model convergence. They subsequently also have much higher Bulk and Tail-ESS as compared to the exact pcGP, pdGP and dGP. Overall, based on the diagnostics, the pcHSGPs and pdHSGPs show much more consistent and stable convergence as compared to their exact versions. Thus, the pcHSGP and pdHSGPs are suggested both in terms of model convergence diagnostics and practicality in applications to large sample data scenarios.

C.4: Summary methods

To evaluate the accuracy of estimating latent variables, we compare posterior samples of the latent input variable $\mathbf{x}$ denoted by $\mathbf{x}_{post}$ from each model with their respective ground truth values denoted by $\mathbf{x}_{true}$. Using $\text{RMSE}(\mathbf{x}_{post}) = \sqrt{\mathbb{E}\left((\mathbf{x}_{post} - \mathbf{x}_{true})^2\right)}$, we look at the combined check of bias-variance trade-off in estimating the latent variables. We compute the posterior RMSE from all fitted models under each simulation scenarios (two scenarios with sample size

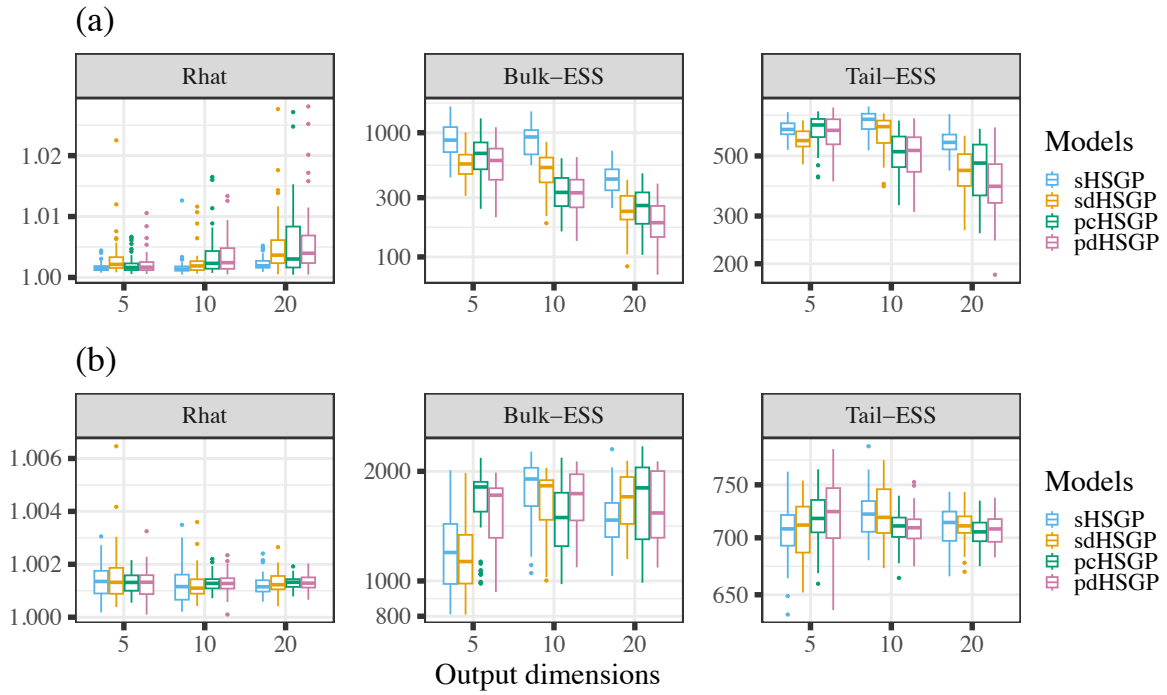

Figure C3: *dGP data scenario: Convergence check for (a) latent inputs and (b) GP hyperparameters of the HSGPs with $N = 100$. The y-axes for Bulk and Tail ESS plots are log10 transformed.*

$N = 20$ and one with $N = 100$) with output dimensions ($D = 5,\ 10$ and 20). We prefer models that provide both low RMSE indicating posterior mean estimates close to the ground truth as well as high precision.

To analyze the results from our experiments, we use a multilevel analysis of variance model (ANOVA) by using brms [17, 18], which disentangles the various components of our simulation study design. With μ_{resp} and σ_{resp} being the mean and SD of our response variable, we use

$$\begin{aligned} \boldsymbol{\mu}_{resp} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{b} + \sum_{\tau} L_{\mu_\tau}(t_\tau) \\ \boldsymbol{\sigma}_{resp} &= \mathbf{X}\boldsymbol{\eta} + \mathbf{Z}\mathbf{u} + \sum_{\tau} L_{\sigma_\tau}(t_\tau) \end{aligned} \tag{6.13}$$

where $\boldsymbol{\beta},\ \mathbf{b}$ (for $\boldsymbol{\mu}_{resp}$) and $\boldsymbol{\eta},\ \mathbf{u}$ (for $\boldsymbol{\sigma}_{resp}$) are coefficients at the population and group levels with $\mathbf{X}$ and $\mathbf{Z}$ being the corresponding design matrices. The $L_\tau(t_\tau)$ terms denote smooth functions over covariates t fitted via splines. We use these models to summarize the results from our experiments.

In the simulation studies, the population level design matrix $\mathbf{X}$ contains covariates representing the different models (exact or approximated) and number of output dimensions along with their interaction terms. For our pcGP data scenario, we only have two comparing models our simulation scenario: the exact pcGP and approximated pcHSGP. Thus we use a two level factor variable accounting for the effect of model choice. In case of the dGP scenario with $N = 20$, we have a five level factor variable denoting our proposed methods pcHSGP, pdHSGP along with their exact versions pcGP, pdGP and the true data generating process dGP.

In case of the large sample dGP simulation scenario with $N = 100$, we have a four level factor variable denoting the single sHSGP, sdHSGP and the composite pcHSGP and pdHSGP. For all of the above cases, we use a three level factor variable depicting the 5, 10, and 20 output dimensions. Through $\mathbf{Z}$ we account for the group-level dependency structure in the response induced by fitting multiple models to the same simulated dataset. We include a random intercept over datasets as well as corresponding random slopes for the model choices made in the respective simulation scenarios.

Lastly, to capture the non-linear relation of the response to the ground truth t, we introduce thin-plate regression spline [96] terms $L_{\mu_\tau}(t_\tau)$ and $L_{\sigma_\tau}(t_\tau)$. The spline terms accounts for any non-linear relationship of the response with respect to the true parameter values t. To summarize the results of latent variable estimation, we specify posterior RMSE as the response in Section 5.4.3 (Chapter 5). In case of model calibration results, we use the log γ scores as our response which we discuss in the following section.

C.5: Simulation based calibration

Using Simulation-Based Calibration (SBC) [29, 58, 84], we test model calibration for estimating latent inputs $\mathbf{x}$. The test is carried out starting with a model, say, $\mathcal{M}_0$. Then, we generate J datasets $\mathbf{y}_{(j)}, j = 1, \ldots, J$ each of size N from the data generating process that exactly aligns with the model $\mathcal{M}_0$. In other words, each individual dataset $\mathbf{y}_{(j)}$ is generated based on a corresponding model parameter draw $\mathbf{x}_{0_{(j)}}$ from its prior distribution $p(\mathbf{x})$. We sample from the posterior approximation by fitting $\mathcal{M}_0$ to each of the datasets $\mathbf{y}_{(j)}$ thus resulting in J fitted models $\mathcal{M}^{(j)}$ with respective posteriors $p(\mathbf{x} \mid \mathbf{y}_{(j)})$ each having H posterior draws $\mathbf{x}_{(j,h)}$. Using $\mathbf{x}_{0_{(j)}}$ as the ground truth, we then calculate a rank statistic for each univariate posterior quantity $h_p^*(\mathbf{x})$ for a specific parameter by counting the number of posterior draws $h_p^*(\mathbf{x}_{(j,h)})$ that are smaller than $h_p^*(\mathbf{x}_{0_{(j)}})$. The rank statistic $R_{(j)}$ for the model $T_{(j)}$ is then given as

$$R_{(j)} = \sum_{h=1}^{H} \mathbb{I}[h_p^*(\mathbf{x}_{(j,h)}) < h_p^*(\mathbf{x}_{0_{(j)}})]. \tag{6.14}$$

The distribution of these single rank-value per model taken together across all J models is a discrete uniform distribution if the approximate posteriors correspond to the true posteriors. Using this property, we assess the correctness of the posterior approximations by testing the rank distribution for uniformity. If the rank distribution departs from uniformity, it indicates a mismatch in the data generating process, model implementation, the posterior approximations or a combination of these.

SBC checks are carried out using a test [58, 83] based on the probability of γ where we observe the most extreme point on the ECDF under the assumption of uniformity. In the latter case, the test statistic is given by

$$\gamma = 2 \min_{j \in \{1,\ldots,J+1\}} (\min\{\text{Bin}(R_j \mid J, z_j), 1 - \text{Bin}(R_j - 1 \mid J, z_j)\}), \tag{6.15}$$

where z_j is the expected proportion of ranks below j such that $z_j = \frac{j}{J+1}$, R_j is the actual

empirical ranks below j, $\text{Bin}(R_j \mid J, z_j)$ is the CDF of the Binomial distribution with J trials and the probability of success evaluated at R. The calculated γ scores (presented on the logarithmic scale for ease of visualization) are then compared to a threshold value below which we reject uniformity [the log γ score; see 83]. The log γ scores are advantageous in summarizing large number of parameters, different models as well us various simulation conditions. Thus, in this paper, considering the numerous models under various simulation scenarios, we evaluate model calibration using the log γ scores.

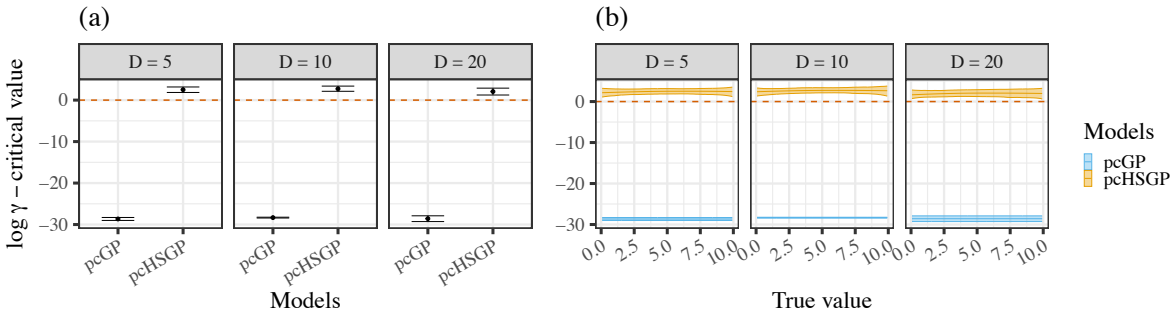

Figure C4: *pcGP data scenario: a) log γ scores offset by the 95% confidence threshold critical value for all the fitted models. The behavior of scores across true latent x values are shown in the right-hand panel. The dashed line denotes the threshold to reject uniformity. b) Shows how the models perform in terms of log γ scores with respect to true values across the input space. Figures are shown for output dimensions $D = 5, 10$ and 20.*

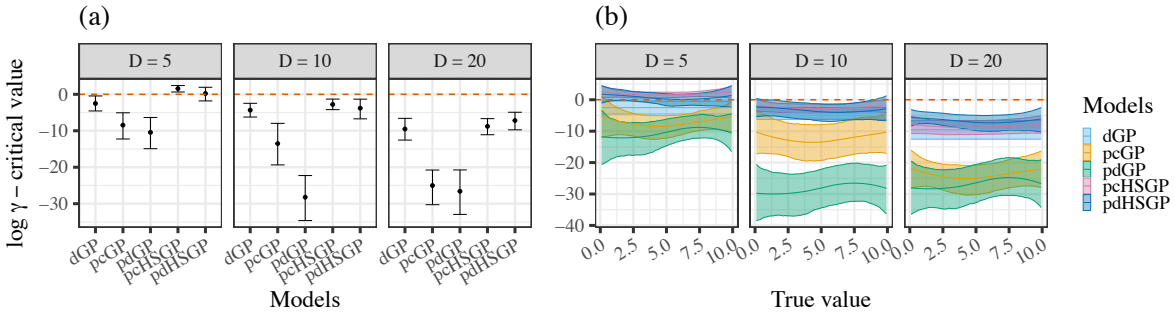

Figure C5: *dGP data scenario: a) log γ scores offset by the 95% confidence threshold critical value for all the fitted models. The behavior of scores across true latent x values are shown in the right-hand panel. The dashed line denotes the threshold to reject uniformity. b) Shows how the models perform in terms of log γ scores with respect to true values across the input space. Figures are shown for output dimensions $D = 5, 10$ and 20.*

To summarize calibration checking results across simulation scenarios, we analyze the log γ scores using a multilevel model [17] described in Appendix C.4 (above). In Figure C4, we show the predicted log γ scores of the exact and approximate models under the pcGP simulation scenario. We clearly see that the pcHSGP yields better calibrated results compared to the exact pcGP despite pcGP being the true data generating process. This result is aligned with the findings in [61] where the latent exact GPs show various degrees of miscalibration presumably due to the interaction of the sampling procedure with the complex posterior geometries of the models.

In Figure C5, we detect lack of calibration across all model choices, however, they are so for a

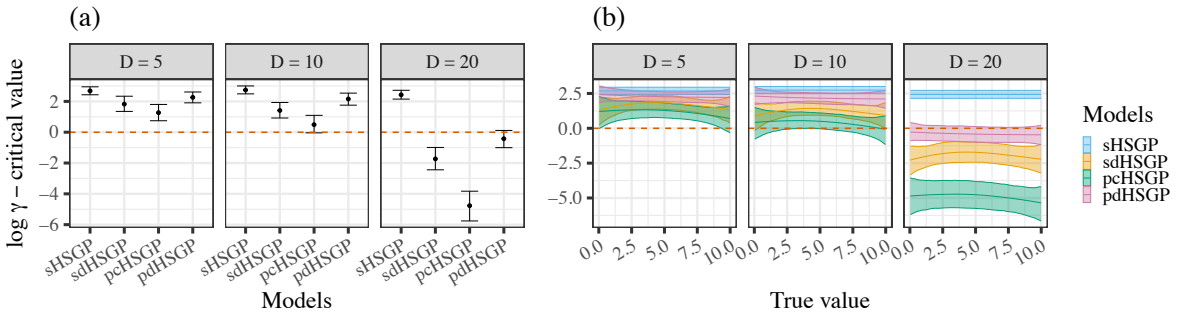

Figure C6: *dGP data scenario: a) log γ scores offset by the 95% confidence threshold critical value for all the fitted models. The behavior of scores across true latent x values are shown in the right-hand panel. The dashed line denotes the threshold to reject uniformity. b) Shows how the models perform in terms of log γ scores with respect to true values across the input space. Figures are shown for output dimensions $D = 5, 10$ and 20.*

few expected reasons.

For dGP, the reason remains the same as the previous case due to the interaction of complex posterior geometry with the sampling method. The struggle of the samplers with these exact models is also evidenced by the comparatively higher $\widehat{R}$ in the convergence diagnostic results (see Appendix C.3 above). For the exact pcGP and pdGP, the miscalibrations additionally are a direct result of relaxing the modeling conditions through using only a partial covariance structure that doesn't fully match the true data generating conditions.

The extreme miscalibrations for pcGP and pdGP in this case are thus due to a combination of this induced model misspecification as well as complex posterior geometry inherent to latent exact GPs. The approximate models pcHSGP and pdHSGP, while not exhibiting the severe miscalibrations of their exact versions, still remain miscalibrated due to the design-induced misspecifications compared to the true data generating process. These miscalibrations however gets rectified to some extent under higher sample sizes as seen in Figure C6. For the higher sample size dGP data scenario where we only compare the HSGPs, all models shows good calibrations except for the higher $D = 20$ case. The frequency of data-model mismatch increase with the number fo output dimensions, thus resulting in worse calibration scores for higher D. Under these conditions, only the sHSGP and pdHSGP (marginally) passes the posterior calibration checks for the dGP data generating scenario. However, considering the worse (by about 50%) latent variable recovery (see Section 5.4.3 Figure 5.3) compared to pdHSGP, the latter remains the only reasonable choice for these specific data generating conditions.

C.6: Hyperparameter recovery

We show the estimation accuracy for hyperparameters of our proposed methods as well as other comparative models involved in each of the simulation scenarios. We, again, present the model evaluation summary using RMSE which combines the model-specific effects on posterior bias and SD for each class of model hyperparameters. We make the comparison based on the class of hyperparameters (for example length-scale) rather than the individual hyperparameters themselves (like $\boldsymbol{\rho}_f$ and $\boldsymbol{\rho}_g$ for composite GPs and $\boldsymbol{\rho}$ for derivative GPs). This conscious choice

is made since comparability remains valid across all models for our simulation scenarios only through the class of hyperparameters and not the individual hyperparameters.

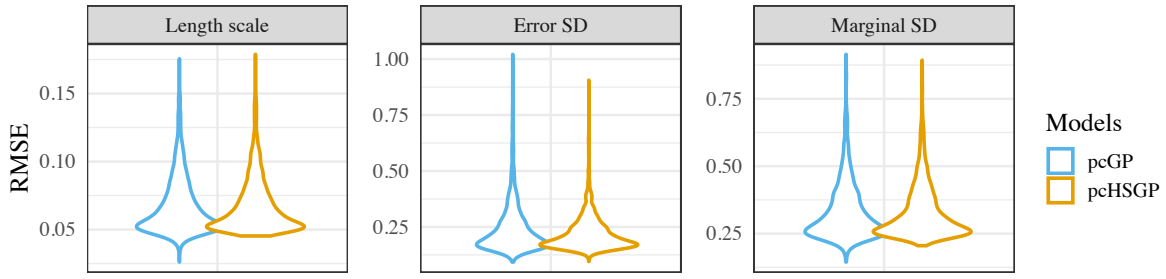

Figure C7: *pcGP data scenario: RMSE on recovery of GP hyperparameters for pcGP and pcHSGP models.*

The $D = 20$ case is arguably the most complex case among the choices of output dimensions we have. Thus, we decided to showcase only for the $D = 20$ since the results were qualitatively same for all the different choices of output dimensions D. For the pcGP data scenario shown in Figure C7, we don't find any clear differences in hyperparameter estimates between the exact pcGP and the approximate pcHSGP. In Figure C8, the approximate pcHSGP and pdHSGP

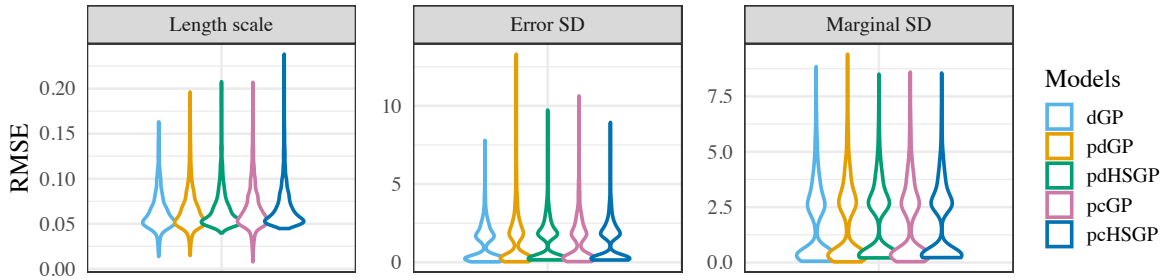

Figure C8: *dGP data scenario: RMSE on recovery of GP hyperparameters for all fitted models.*

registers similar RMSE results for all of the hyperparameters across all models. We note that for length scale and error sd, dGP has lesser extreme RMSEs. We hypothesize that the partial covariance structure for our proposed HSGPs is likely the primary reason for these phenomena.

The bimodality in GP marginal SD and error SDs are likely due to the scale differences between GP functions $\mathbf{f}$, $\mathbf{g}$ and the corresponding observations $\mathbf{y}_f$, $\mathbf{y}_g$ (similarly $\mathbf{f}^{(1)}$ and $\mathbf{y}^{(1)}$ for the derivative cases). As the dGP data generating conditions dictate that the derivative $\mathbf{f}^{(1)}$ and subsequently $\mathbf{y}^{(1)}$ are at a significantly lower scale that $\mathbf{f}$ and $\mathbf{y}$, we see this reflected in the RMSEs of marginal and error SDs which modeling this scale imbalance. This behavior is clearly seen in Figure C9 for the marginal and error SDs in sHSGP and sdHSGP which are fitted to $\mathbf{y}$ and $\mathbf{y}^{(1)}$ separately. Based on all of the results, the HSGPs and their exact models perform equally when it comes to hyperparameter estimation accuracy for every demonstrated simulation

scenario.

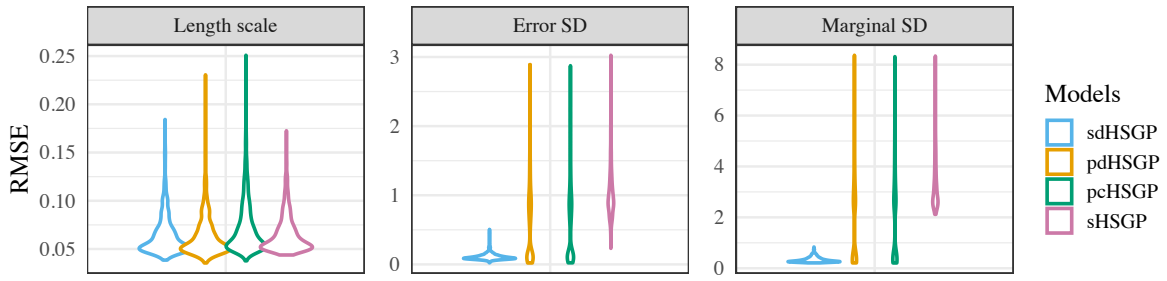

Figure C9: *dGP data scenario ($N = 100$): RMSE on recovery of GP hyperparameters for all fitted models.*

C.7: Additional case study results

We present additional results to the case study shown in Section 5.5 (Chapter 5) using a smaller set of genes. We specifically select 5 genes with highly variable gene expression levels (with a higher degree of non-linearity with respect to experimental time). The list of genes used in the

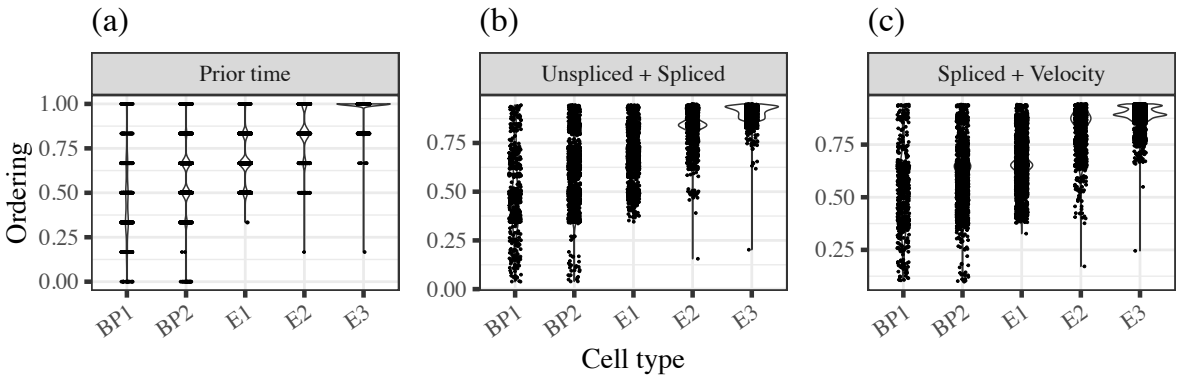

Figure C10: *Case study with 5 genes: a) Distribution of discrete experimental time by cell type. b) Posterior latent continuous cell ordering from modeling unspliced and spliced gene expression using pcHSGP. c) Posterior latent continuous cell ordering from modeling spliced gene expression and RNA velocity using pdHSGP. The x-axis denotes the cell types blood progenitors (BP) and erythroid (E).*

full case study and this shorter study respectively are presented below. This shorter version of case study with $N = 9815$ and $D = 5$ shows qualitatively similar results as the larger study based on the full $D = 14$ genes suggested in [7]. Figures C10 and C11 demonstrate quantitatively similar results as the larger case study. The model fitting times are, however, considerably shorter with pcHSGP and pdHSGP taking 12 and 22 hrs respectively on the same computing resources mentioned in Section 5.5.

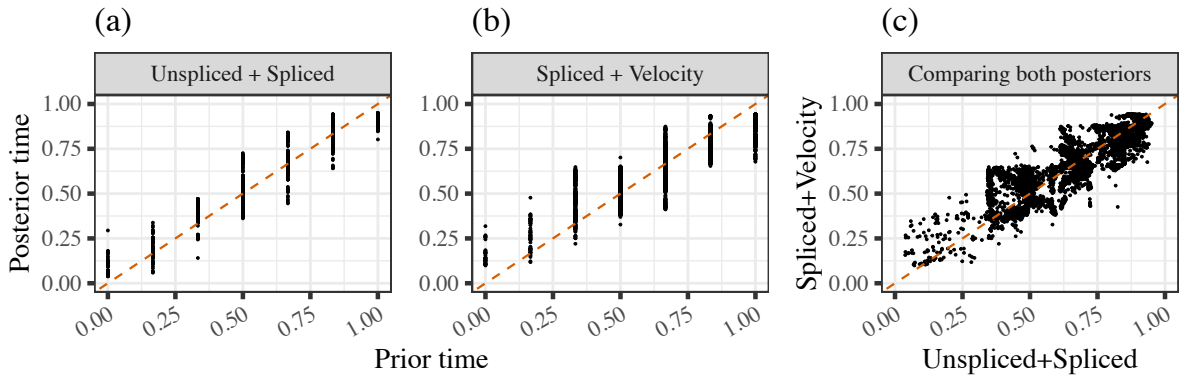

Figure C11: *Case study with 5 genes: Deviations of posterior latent cell ordering from discrete experimental times based on a) modeling unspliced and spliced gene expression using pcHSGP and b) modeling spliced gene expression and RNA velocity using pdHSGP. c) Comparison of posterior latent cell orderings from both the models.*

Table C1: List of gene names that were involved in the cases studies.

Gene names	Full case study	Short case study
Blvrb	✓	✓
Coro2b	✓	
Hba.x	✓	
Hbb.y	✓	
Mllt3	✓	
Myo1b	✓	
Nfkb1	✓	
Phc2	✓	
Rbms2	✓	✓
Skap1	✓	✓
Smarca2	✓	
Smim1	✓	✓
Sulf2	✓	
Yipf5	✓	✓