Learning GPLVM with arbitrary kernels using the unscented transformation

Daniel Augusto de Souza[1], **Diego Mesquita**[2], **César Lincoln Mattos**[1], **João Paulo Gomes**[1]
[1]Federal University of Ceará, [2]Aalto University
dani@spectral.space, diego.mesquita@aalto.fi, cesarlincoln@dc.ufc.br, jpaulo@dc.ufc.br

Abstract

Gaussian Process Latent Variable Model (GPLVM) is a flexible framework to handle uncertain inputs in Gaussian Processes (GPs) and incorporate GPs as components of larger graphical models. Nonetheless, the standard GPLVM variational inference approach is tractable only for a narrow family of kernel functions. The most popular implementations of GPLVM circumvent this limitation using quadrature methods, which may become a computational bottleneck even for relatively low dimensions. For instance, the widely employed Gauss-Hermite quadrature has exponential complexity on the number of dimensions. In this work, we propose using the unscented transformation instead. Overall, this method presents comparable, if not better, performance than off-the-shelf solutions to GPLVM, and its computational complexity scales only linearly on dimension. In contrast to Monte Carlo methods, our approach is deterministic and works well with quasi-Newton methods, such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm. We illustrate the applicability of our method with experiments on dimensionality reduction and multistep-ahead prediction with uncertainty propagation.

1 INTRODUCTION

Gaussian process (GP) models have been widely adopted in the machine learning community as a Bayesian approach to nonparametric kernel-based learning due to their simplicity and fully probabilistic predictions (Rasmussen and Williams 2006). Thanks to its flexibility, many authors have applied the GP framework in contexts such as dynamical modeling (Eleftheriadis, Nicholson, et al. 2017; Mattos et al. 2016), autoencoders (Casale et al. 2018; Eleftheriadis, Rudovic, et al. 2017), and hierarchical modeling (Havasi et al. 2018; Salimbeni and Deisenroth 2017).

The works above rely on a common building block: the GP Latent Variable Model (GPLVM), proposed by Lawrence (2004) to handle learning scenarios with uncertain inputs. The GPLVM was extended with a Bayesian training approach (Bayesian GPLVM) by Titsias and Lawrence (2010) and later by Damianou and Lawrence (2013) in a multilayer setting (Deep GPs).

The variational approach presented by Titsias and Lawrence (2010) for the Bayesian GPLVM presents tractable calculations only for a few choices of kernel function, such as the radial basis function (RBF) kernel. However, the RBF kernel presents limited extrapolation capability (MacKay 1998). Some authors have tried to address that issue. The work by Duvenaud et al. (2013) and Lloyd et al. (2014) pursues a compositional approach to building more expressive kernels from simpler ones. Wilson and Adams (2013) propose the spectral mixture kernel family, capable of automatic pattern discovery and extrapolating beyond the training data. Al-Shedivat et al. (2017), Wilson, Hu, et al. (2016a), and Wilson, Hu, et al. (2016b) propose using deep neural networks to learn kernel functions directly from the available data. Although those proposals achieve more flexible models than those with the RBF kernel, they turn some Bayesian GPLVM expressions intractable.

Alternatively, some works (e.g. Eleftheriadis, Nicholson, et al. 2017; Salimbeni and Deisenroth 2017) handle non-RBF kernels with uncertain inputs using the so-called "reparametrization trick" (Kingma and Welling 2014; Rezende et al. 2014) in the doubly stochastic variational inference framework, intro-

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130. Copyright 2021 by the author(s).

duced by Titsias and Lázaro-Gredilla (2014). This approach results in a flexible methodology, but, in contrast to deterministic inference, it does not support popular quasi-Newton optimizers, like the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm.

In this work, we handle the propagation of uncertainty in the GPLVM while maintaining the non-stochastic framework presented by Titsias and Lawrence (2010). We tackle the intractabilities of uncertain inputs and non-RBF kernels by employing the unscented transformation (UT), a deterministic technique to approximate nonlinear mappings of a random variable (Julier and Uhlmann 2004; Menegaz et al. 2015). The UT projects a finite number of *sigma points* through this mapping and uses the output statistics to estimate the mean and covariance of the transformed random variable, resulting in a more scalable method than, for instance, the Gauss-Hermite (GH) quadrature.

We use the UT to handle the intractabilities of the Bayesian GPLVM and propose using this approximation in the integrals that arise by convolving kernel functions and a Gaussian density in the variational framework by Titsias and Lawrence (2010). Our methodology enables the use of any kernel, including ones obtained via auxiliary parametric models in a kernel learning setup, while maintaining fast deterministic inference and without imposing any additional hyperparameters. We evaluate this approach in GPLVM's original task of dimensionality reduction in the uncertainty propagation during a free simulation (multistep-ahead prediction) of dynamical models. Our experimental results show that, even for a moderate latent space size, the commonly used GH quadrature is only feasible when the user picks a very low number of evaluation points. Moreover, in such scenarios, the UT still presents excellent results.

In summary, our main contributions are: *i*) an extension to the Bayesian GPLVM using the UT to handle intractable integrals deterministically and enable the use of any kernel; *ii*) a set of experiments comparing the proposed approach and alternative approximations using Gauss-Hermite quadrature and Monte Carlo sampling in tasks involving dimensionality reduction and dynamical free simulation.

We organize the remainder of this paper as follows. Section 2 presents the theoretical background by summarizing the GPLVM framework and the UT approximation. In Section 3, we detail our proposal to apply the UT within the Bayesian GPLVM setting. In Section 4, we present and discuss the obtained empirical results. Finally, in Section 5, we review related works related to GPs and UT, and we conclude the paper in Section 6 with ideas for further work.

2 THEORETICAL BACKGROUND

In this section, we summarize GPs, the Bayesian GPLVMs, and the UT.

2.1 The Gaussian Process Framework

Let N inputs $\boldsymbol{x}_i \in \mathbb{R}^{D_x}$, organized in a design matrix $\boldsymbol{X} \in \mathbb{R}^{N \times D_x}$ be mapped via $f : \mathbb{R}^{D_x} \to \mathbb{R}^{D_y}$ to N correspondent outputs $\boldsymbol{f}_i \in \mathbb{R}^{D_y}$, organized in the matrix $\boldsymbol{F} \in \mathbb{R}^{N \times D_y}$. We observe $\boldsymbol{Y} \in \mathbb{R}^{N \times D_y}$, a noisy version of $\boldsymbol{F}$. Considering an observation noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$, we have $\boldsymbol{f}_{:d} = [f(\boldsymbol{x}_1) \dots f(\boldsymbol{x}_N)]^\top$ and $\boldsymbol{y}_{:d} = \boldsymbol{f}_{:d} + \boldsymbol{\epsilon}$, where $\boldsymbol{y}_{:d} \in \mathbb{R}^N$ is comprised of the d-th component of each observed sample, i.e., the d-th column of the matrix $\boldsymbol{Y}$. If we choose independent multivariate zero mean Gaussian priors for each dimension of $\boldsymbol{F}$, we get (Rasmussen and Williams 2006):

$$p(\boldsymbol{Y}|\boldsymbol{X}) = \prod_{d=1}^{D_y} \mathcal{N}(\boldsymbol{y}_{:d}|\boldsymbol{0}, \boldsymbol{K}_f + \sigma^2 \boldsymbol{I}),$$

where we were able to analytically integrate out the non-observed (*latent*) variables $\boldsymbol{f}_{:d}|_{d=1}^{D_y}$. The elements of the covariance matrix $\boldsymbol{K}_f \in \mathbb{R}^{N \times N}$ are calculated by $[\boldsymbol{K}_f]_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j), \forall i, j \in \{1, \cdots, N\}$, where $k(\cdot, \cdot)$ is the so-called covariance (or *kernel*) function.

2.2 The Bayesian GPLVM

The Gaussian Process Latent Variable Model (GPLVM), proposed by Lawrence (2004), extends the GP framework for scenarios where we do not observe the inputs $\boldsymbol{X}$, which generated the response variables $\boldsymbol{Y}$ via the modeled function. The GPLVM was initially proposed in the context of nonlinear dimensionality reduction[1], which can be done by choosing $D_x < D_y$. However, the approach has proved to be flexible enough to be used in several other scenarios. For instance, in supervised tasks, the matrix $\boldsymbol{X}$ can be seen as a set of observed but uncertain inputs (Damianou, Titsias, et al. 2016).

The Bayesian GPLVM, proposed by Titsias and Lawrence (2010), considers a variational approach (Jordan et al. 1999) to approximately integrate the latent variables $\boldsymbol{X}$. Inspired by Titsias' variational sparse GP framework (Titsias 2009), the Bayesian GPLVM avoids overfitting by considering the uncertainty of the latent space and enables the determination of D_x by using a kernel function with ARD (*automatic relevance determination*) hyperparameters.

Following Titsias and Lawrence (2010), we start by including M inducing points $\boldsymbol{u}_{:d} \in \mathbb{R}^M$ associated to

[1] The GPLVM is a nonlinear extension of the probabilistic Principal Component Analysis (Lawrence 2004).

each output dimension and evaluated in M pseudo-inputs $\boldsymbol{z}_j|_{j=1}^{M} \in \mathbb{R}^{D_x}$, where $p(\boldsymbol{u}_{:d}) = \mathcal{N}(\boldsymbol{u}_{:d}|\boldsymbol{0}, \boldsymbol{K}_u)$ and $\boldsymbol{K}_u \in \mathbb{R}^{M\times M}$ is the kernel matrix computed from the pseudo-inputs. The joint distribution of all the variables in the GPLVM $p(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{F}, \boldsymbol{U})$ is now given by (with omitted dependence on $\boldsymbol{z}_j$):

$$p(\boldsymbol{X}) \prod_{d=1}^{D_y} p(\boldsymbol{y}_{:d}|\boldsymbol{f}_{:d}) p(\boldsymbol{f}_{:d}|\boldsymbol{u}_{:d}, \boldsymbol{X}) p(\boldsymbol{u}_{:d}).$$

Then, we define an approximation to the posterior $q(\boldsymbol{X}, \boldsymbol{F}, \boldsymbol{U}) = \prod_{d=1}^{D_y} q(\boldsymbol{x}_{:d}) q(\boldsymbol{u}_{:d}) p(\boldsymbol{f}_{:d}|\boldsymbol{u}_{:d}, \boldsymbol{X})$ with a mean-field approximation for $q(\boldsymbol{x}_{:d}) = \prod_{i=1}^{N} q(x_{id})$.

By applying Jensen's inequality to the joint distribution of each output dimension, we obtain the evidence lower bound (ELBO):

$$\begin{aligned}\log p(\boldsymbol{y}_{:d}) \geq \; & \frac{\ln|\boldsymbol{K}_u|}{2} - \frac{n\ln\left(2\pi\sigma^2\right)}{2} - \frac{\ln|\sigma^2\boldsymbol{K}_u + \boldsymbol{\Psi}_2|}{2} \\ & - \frac{\boldsymbol{y}_{:d}^\intercal \boldsymbol{y}_{:d}}{2\sigma^2} + \frac{\boldsymbol{y}_{:d}^\intercal \boldsymbol{\Psi}_1 \left(\sigma^2 \boldsymbol{K}_u + \boldsymbol{\Psi}_2\right)^{-1} \boldsymbol{\Psi}_1^\intercal \boldsymbol{y}_{:d}}{2\sigma^2} \\ & - \frac{\psi_0}{2\sigma^2} + \frac{\operatorname{tr}\left(\boldsymbol{K}_u^{-1}\boldsymbol{\Psi}_2\right)}{2\sigma^2},\end{aligned}$$

in which ψ_0, $\boldsymbol{\Psi_1}$ and $\boldsymbol{\Psi_2}$ involve convolutions of the kernel function with the variational distribution. These values are known as Ψ-statistics and are tractable only for a few kernel functions, such as the RBF, the linear kernels, and their mixtures. For completeness, we provide a detailed derivation in the supplementary material.

2.3 The Unscented Transformation

The unscented transformation (UT) is a method for estimating the first two moments of a transformed random variable under an arbitrary function. First proposed by Uhlmann (1995) for nonlinear Kalman filters, the transformation itself is decoupled from the proposed Unscented Kalman Filter.

The UT approximates the mean and covariance of the transformed random variable with a weighted average of transformed *sigma points* $\boldsymbol{S}$, derived from the first two moments of the original input.

Let $p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{x} \in \mathbb{R}^D$, be the input of an arbitrary transformation $\boldsymbol{f}\colon \mathbb{R}^D \to \mathbb{R}^Q$. Given uniform weights for the sigma points, the output moments are computed by:

$$\begin{aligned}\langle \boldsymbol{f}(\boldsymbol{x})\rangle_{p(\boldsymbol{x})} &\approx \frac{\sum_{i=1}^{2D} \boldsymbol{f}(\boldsymbol{s}_i)}{2D} = \tilde{\boldsymbol{\mu}}_{\text{UT}}, \qquad (1)\\ \mathbf{Cov}(\boldsymbol{f}(\boldsymbol{x})) &\approx \frac{\sum_{i=1}^{2D} (\boldsymbol{f}(\boldsymbol{s}_i) - \tilde{\boldsymbol{\mu}}_{\text{UT}})(\boldsymbol{f}(\boldsymbol{s}_i) - \tilde{\boldsymbol{\mu}}_{\text{UT}})^\intercal}{2D}.\end{aligned}$$

There are several strategies to select sigma points[2]. However, we follow the original scheme by Uhlmann (1995), with uniform weights and sigma points chosen from the columns of the squared root of $D\boldsymbol{\Sigma}$, an efficient way to generate a symmetric distribution of sigma points.

This scheme is defined as follow. Let $\mathbf{Chol}(\boldsymbol{\Sigma})$ be the Cholesky decomposition of the matrix $\boldsymbol{\Sigma}$. Then, the sigma points $\boldsymbol{S}$ are defined as:

$$\begin{aligned}\boldsymbol{s}_i &= \boldsymbol{\mu} + [\mathbf{Chol}(D\boldsymbol{\Sigma})]_{:i}\\ \boldsymbol{s}_{i+D} &= \boldsymbol{\mu} - [\mathbf{Chol}(D\boldsymbol{\Sigma})]_{:i}, \quad \forall i \in [1, D],\end{aligned}$$

where $[\mathbf{Chol}(D\boldsymbol{\Sigma})]_{:i}$ denotes the i-th column of the matrix $\mathbf{Chol}(D\boldsymbol{\Sigma})$.

3 PROPOSED METHODOLOGY

This section details our proposal, discusses its advantages and limitations, and presents an initial empirical validation.

3.1 Learning Bayesian GPLVMs using UT

As mentioned in Section 2.2, the computation of the Ψ-statistics is the only part that hinders the application of Bayesian GPLVMs with arbitrary kernels. For the intractable cases, we propose cases using the mean provided by the UT (Equation 1) as follows:

$$\begin{aligned}\psi_0 &= \sum_{i=1}^{N} \langle k(\boldsymbol{x}_i, \boldsymbol{x}_i)\rangle_{q(\boldsymbol{x}_i)} \qquad (2)\\ &\approx \frac{1}{2D_x} \sum_{i=1}^{N} \sum_{k=1}^{2D_x} k(\boldsymbol{s}_k^{(i)}, \boldsymbol{s}_k^{(i)}),\end{aligned}$$

$$[\boldsymbol{\Psi}_1]_{ij} = \langle k(\boldsymbol{x}_i, \boldsymbol{z}_j)\rangle_{q(\boldsymbol{x}_i)} \approx \frac{1}{2D_x} \sum_{k=1}^{2D_x} k(\boldsymbol{s}_k^{(i)}, \boldsymbol{z}_j), \qquad (3)$$

$$\begin{aligned}[\boldsymbol{\Psi}_2]_{jm} &= \sum_{i=1}^{N} \langle k(\boldsymbol{x}_i, \boldsymbol{z}_j) k(\boldsymbol{x}_i, \boldsymbol{z}_m)\rangle_{q(\boldsymbol{x}_i)} \qquad (4)\\ &\approx \frac{1}{2D_x} \sum_{i=1}^{N} \sum_{k=1}^{2D_x} k(\boldsymbol{s}_k^{(i)}, \boldsymbol{z}_j) k(\boldsymbol{s}_k^{(i)}, \boldsymbol{z}_m),\end{aligned}$$

where $\psi_0 \in \mathbb{R}$, $\boldsymbol{\Psi}_1 \in \mathbb{R}^{N\times M}$, and $\boldsymbol{\Psi}_2 \in \mathbb{R}^{M\times M}$ are the Ψ-statistics and $\boldsymbol{s}_k^{(i)}$ indicates the k-th sigma point related to $q(\boldsymbol{x}_i)$.

3.2 Advantages and limitations

Besides enabling the use of non-analytical kernels in the Bayesian GPLVM, the choice of using UT-based

[2] e.g. Menegaz et al. (2015)

approximations in place of, for instance, the Gauss-Hermite (GH) quadrature, brings great computational benefits due to the number of points evaluated to compute the Gaussian integral. Given a D-dimensional random variable, the UT requires just a linear number of $2D$ evaluations. In contrast, the GH quadrature requires H^D evaluations, where H is a user-chosen order parameter. Even for $H = 2$ and moderate dimensionality values, e.g. $D = 20$, the GH approach would require at least 2^{20} evaluations per approximation, which is infeasible.

Since an exponentially lower number of function evaluations is required, the UT presents a practical alternative to the GH quadrature. Furthermore, since the sigma points are obtained in a fully deterministic manner, it enables quasi-Newton optimization methods, unlike Monte Carlo integration. Nevertheless, if a large quantity of evaluations is allowed for either GH quadrature or Monte Carlo (MC) integration, in exchange for additional computational effort, it is expected that the lower number of sigma points of the UT would result in a coarser approximation.

Regarding the approximation quality, Menegaz et al. (2015) proved that our choice for sigma points (detailed in Section 2.3) enables computing the projected mean correctly up to the third-order Taylor series expansion of the transformation function if $\boldsymbol{x}$ is Gaussian distributed. Note that the GH quadrature approximation is guaranteed up to the $(2H-1)$th order of the function. So, for the $H = 2$ case, it is expected that both approximations will have about the same quality.

3.3 Preliminary validation

In the Bayesian GPLVM, the amount of sampled points is relevant since the approximations are computed at each step of the variational lower bound optimization. Thus, the number of times we evaluate the Ψ-statistics gives a raw estimate of the chosen approximation computational budget.

To verify how performance evolves with dimensionality when using UT in the context of the Bayesian GPLVM, we computed $\boldsymbol{\Psi}_1$ – see Equation 3 – considering an RBF kernel on random data ($N = 40, M = 20$) of varying dimension. We compare the UT result with the GH quadrature and MC integration.

Figure 1 shows a comparison of the relative time spend between the UT and four competing quadrature methods: GH and MC with $2D$, 2^D, and 200 samples. As expected from the theoretical complexity, GH's exponential nature makes it infeasible at dimensions beyond 10. Not only that, but the complexity of GH brings an additional overhead that is apparent even in small dimensions. On the other hand, MC integration's simplicity brings its runtime to be faster than UT on small dimensions, but, as it will become apparent in the next section, at this regime, the UT can achieve better results even when compared to the MC with 200 samples.

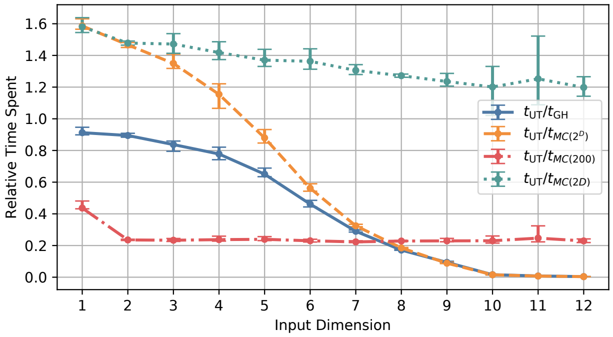


Figure 1: Relative computational effort of the UT with other methods when computing Ψ-statistics. In practice, even in low dimensions, the overhead of computing the Hermite roots and weights can make GH slower than UT. As expected, the exponential nature of GH makes it infeasible at dimensions beyond 10.

4 EXPERIMENTS

This section intends to evaluate the UT against other approximations methods, showing its practicability in quality and speed on tasks requiring solving Psi-statistics during model training and model prediction. We considered two standard tasks for the GPLVM that fit this criterion: dimensionality reduction and free simulation of dynamical models with uncertainty propagation.

We compared the proposed UT approach with the GH quadrature and the reparametrization trick based MC sampling for computing the Ψ-statistics of the Bayesian GPLVM. In the tractable cases, we also considered the analytical expressions. All experiments were implemented in Python using the GPflow framework (G. Matthews et al. 2017). The code is available at `https://github.com/spectraldani/UnscentedGPLVM/`.

To maintain a reasonable computational cost for the GH experiments, we used 2^D points, where D is the input dimension. For the MC approximations, we used three different numbers of samples: *i*) the same as UT, *ii*) the same as GH, and *iii*) a fixed quantity of 200 samples. Each MC experiment was run ten times, with averages and standard deviations reported. The MC approximation is similar to the one in the doubly stochastic variational framework (Titsias and Lázaro-Gredilla 2014), but without mini-batch updates.

The kernel hyperparameters, likelihood noise, and variational parameters are all jointly optimized using the second-order optimization method L-BFGS-B (Byrd et al. 1995). However, it is not feasible to use L-BFGS-B for the models with MC sampling, so these models were optimized using Adam (Kingma and Welling 2014) with a learning rate of 0.01.

4.1 Dimensionality Reduction

The dimensionality reduction task is especially suitable for the UT-based approach since the dimension of the integrand in the Ψ-statistics are usually small for data visualization purposes.

We used two datasets, which were referred in Lawrence (2004) and Titsias and Lawrence (2010), the Oil flow dataset, and the USPS digit dataset. In both cases, we compared the analytic Bayesian GPLVM model with the RBF kernel against a kernel with non-analytic Ψ-statistics. The following kernels were considered: RBF, Matérn 3/2, and a Multilayer Perceptron (MLP) composed on an RBF kernel, similar to the manifold learning approach by Calandra et al. (2016).

The variational means were initialized based on standard Principal Component Analysis (PCA), and the latent variances were initialized to 0.1. Also, 20 points from the initial latent space were selected as inducing pseudo-inputs and were appropriately optimized during training.

Each scenario was evaluated following two approaches: a qualitative analysis of the learned two-dimensional latent space, a quantitative metric in which we took the known labels from each dataset and computed the predictive accuracy of the predicted classes of points in the latent space. In the latter, we used a five-fold cross-validated 1-nearest neighbor (1-NN). For the quantitative results, we also show the accuracy of the PCA projection for reference.

4.1.1 Oil Flow Dataset

The multiphase Oil flow dataset consists of 1000 observations with 12 attributes, belonging to three classes (Bishop and James 1993). We applied GPLVM with five latent dimensions and selected the two dimensions with the greatest inverse lengthscales.

For the approximations with the GH quadrature, we used $2^5 = 32$ samples. This contrasts with the UT, which only uses $2 \cdot 5 = 10$ samples. Note that we have attempted to follow Titsias and Lawrence (2010) and use ten latent dimensions, but that would require the GH to evaluate $2^{10} = 1024$ samples at each optimization step, which made the method too slow on the tested hardware. On the other hand, since UT scales linearly with the integrand dimension, it was still feasible and provided consistent results.

Table 1: 1-NN accuracies results for the Oil flow dataset. Note that the UT managed to achieve better results while using $\frac{1}{3}$ of the evaluations as GH.

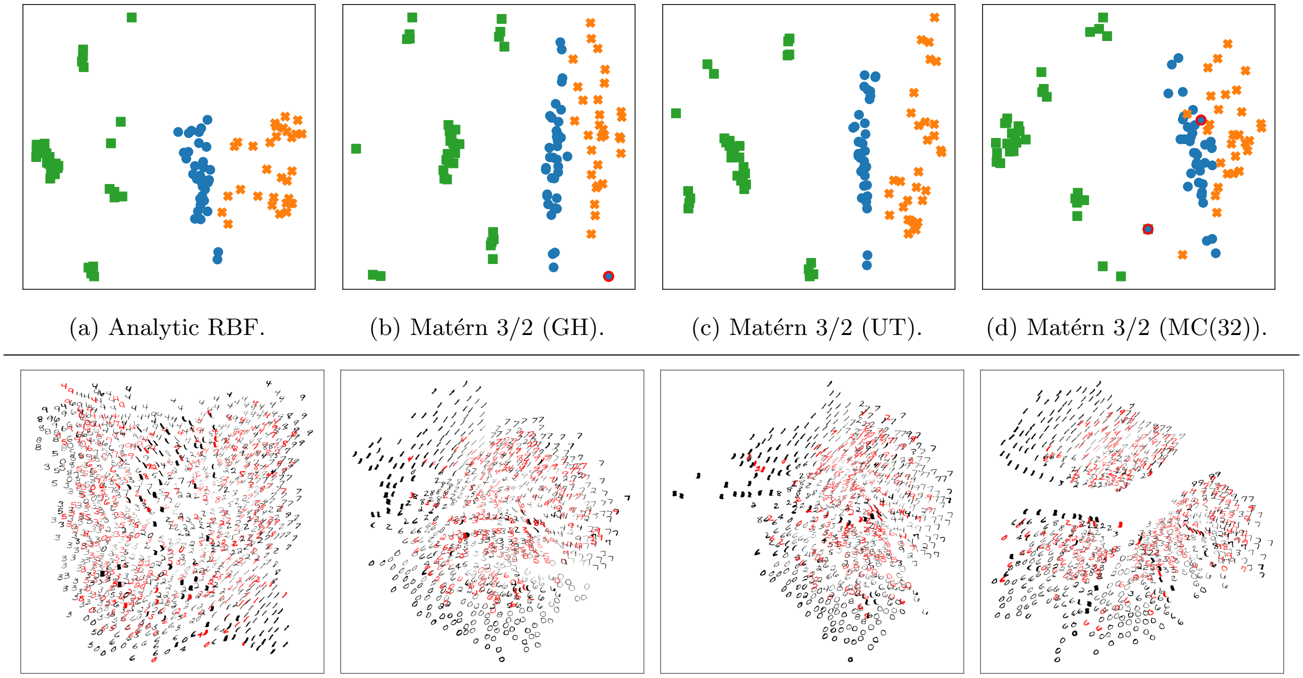
Method	# evaluations	Kernel	Accuracy
PCA	-	-	79.0 ± 6.5
Analytic	-	RBF	98.0 ± 2.7
Gauss-Hermite	32	Matérn 3/2	95.0 ± 6.1
		RBF	98.0 ± 2.7
Unscented	10	Matérn 3/2	100.0 ± 0.0
		RBF	98.0 ± 2.7
Monte Carlo	10	Matérn 3/2	85.6 ± 8.7
		RBF	98.2 ± 2.4
	32	Matérn 3/2	87.9 ± 5.4
		RBF	98.0 ± 2.5
	200	Matérn 3/2	95.4 ± 3.0
		RBF	97.0 ± 4.0

Figure 2 shows that independent of the chosen method to solve the Ψ-statistics, either the analytic expressions or any of the deterministic approximations yield similar overall qualitative results. Table 1 contains statistics of the 1-NN accuracies for all kernels and approximation methods. As expected, all the nonlinear approaches performed better than regular PCA. The RBF results for the deterministic strategies are identical, while the Matérn 3/2 kernel with the UT approximation obtained slightly better results overall. However, when using MC estimates with the same amount of points that UT and GH used and the Matérn 3/2 kernel, the results were worse than both UT and GH.

4.1.2 USPS Digit Dataset

The USPS digit dataset contains 7000 16 × 16 grayscale images of handwritten numerals from 0 to 9. To soften the required computational effort, we used just 500 samples of each class. We used a GPLVM with five latent dimensions on all kernels except the MLP kernel, where two latent dimensions were used. The same evaluation methodology previously described was followed. Additionally, we ran this experiment ten times.

We expected the MLP kernel to fare better than the RBF kernel due to neural networks' well-known capabilities to find lower-dimensional representations of higher dimensional structured data (Wilson, Hu, et al. 2016b). From Table 2, this was the case since all methods had an increase of 30% accuracy compared to their results with RBF. We also noted that even MC approximations with more evaluations than UT and GH do not achieve the same results.



(a) Analytic RBF. (b) Matérn 3/2 (GH). (c) Matérn 3/2 (UT). (d) Matérn 3/2 (MC(32)).

(e) Analytic RBF. (f) MLP kernel (GH). (g) MLP kernel (UT). (h) MLP kernel (MC(200)).

Figure 2: Projections of the Oil flow and USPS digits datasets for GPLVM with different kernels and approximations. The projections shown are the ones with median score obtained in the cross-validation steps. 1-NN mislabels are marked in red. By visual inspection, MC approximations deviated the most from the other approximations despite having the same model as the others.

Figure 2 compares the analytic solution with RBF versus the approximate solutions using the MLP kernel with a single hidden layer and [2, 30, 60] neurons (input, hidden, and output, respectively). Visually, the difference between the kernels is as stark, as noted in the quantitative results. These plots also show that the MC approximation finds a very different projection than the other methods that are arguably more difficult to interpret due to the appearance of a gap in the latent data.

Table 2: 1-NN accuracies results for the USPS dataset. The use of a more complex kernel brought benefits to all methods. Despite its simplicity, the UT has better or similar results on all kernels.

Method	# evaluations	Kernel	Accuracy
PCA	-		35.6 ± 1.0
Analytic	-	RBF	36.7 ± 1.4
Gauss-Hermite	4	MLP	69.0 ± 1.2
	32	RBF	36.4 ± 1.6
Unscented	4	MLP	68.8 ± 1.3
	10	RBF	39.5 ± 1.6
Monte Carlo	4	MLP	47.7 ± 1.7
	10	RBF	26.8 ± 1.6
	32	RBF	27.3 ± 1.4
	200	MLP	54.1 ± 1.8
		RBF	29.2 ± 1.3

4.2 Dynamical Free Simulation

Free simulation, or multistep-ahead prediction, is a task that consists of forecasting the values of a dynamical system arbitrarily far into the future based on past predicted values. In most simple models, such as the GP-NARX (Kocijan et al. 2005), each prediction does not depend on the uncertainty of past predictions but only past mean predicted values. The lack of dependency between the current forecasts and the uncertainty of past predictions can be a significant problem because the user cannot be confident about the quality of the prediction if it does not consider the compounded errors from past estimates.

To propagate the uncertainty of each prediction to the next implies performing predictions with uncertain inputs. This task has been tackled before, for instance, by Girard et al. (2003), but for GP models using the RBF kernel.

In this experiment, we first trained a GP-NARX without considering uncertain inputs, following the vanilla NARX approach (Kocijan et al. 2005). Then, we applied the same optimized kernel hyperparameters in a GPLVM, selecting all the training inputs as pseudo-inputs. Finally, the GPLVM is used to perform a free simulation with uncertain inputs formed by the past predictive distributions. Since we applied approximations for computing the Ψ-statistics in the predictions, any proper kernel function can be chosen.

4.2.1 Airline Passenger Dataset

The Airline passenger numbers dataset records monthly passenger numbers from 1949 to 1961 (Hyndman 2018). We used the first four years for training and left the rest for testing, and chose an autoregressive lag of 12 past observations as input. After the GP-NARX kernel hyperparameters are optimized, we choose the variance of the variational distribution in the GPLVM to be equal to the optimized noise variance. The free simulation starts from the beginning of the training set until the end of the test set, using past predicted variances as variational variances of the uncertain inputs, enabling approximate uncertainty propagation during the simulation.

We used the following kernels: a mixture of an RBF kernel with a linear kernel, a mixture of periodic[3], RBF, and linear kernels. We choose the latter combination of kernels because of our prior knowledge that airplane ticket sales follow a periodic trend and have an overall upward tendency because of the popularity increase and decrease in ticket prices. We emphasize that the choice of such a flexible combination of kernels would not be possible without using approximate methods when considering the uncertain inputs scenario and the GPLVM framework.

Quantitative evaluation is done by computing the RMSE, given by $\sqrt{\frac{1}{n^*}\sum_i^{n^*}(y_i - \mu_i^*)^2}$, where n^* is the number of test samples, y_i is the true output, and μ_i^* is the predicted mean output. The average NLPD is also used as an evaluation metric, and it is defined as $\frac{1}{2}\log 2\pi + \frac{1}{2n^*}\sum_i^{n^*}\left[\log \sigma_i^{*2} + \frac{(y_i-\mu_i^*)^2}{\sigma_i^{*2}}\right]$, where σ_i^{*2} is the i-th predicted variance. The RMSE and average NLPD are computed using the test set. For both metrics, lower is better.

Table 3 presents the obtained results. Although with similar RMSE, all GPLVM variants showed better NLPD values than their standard GP-NARX counterparts. That is expected since the uncertainty of each prediction is being approximately propagated to the next predictions. As for the models with UT, its results were better than the equivalent MC sample sizes but had a much better cost-benefit over the other methods given that they are using 8 to 170 times more samples for a 0.07 to 0.06 decrease in NLPD. As shown in Figure 3, the visual difference between the two methods is subtle.

Table 3: Summary of the free simulation results for the Air passengers dataset. All methods have comparable RMSE but when comparing with GH, a 170 fold increase in evaluations resulted with just a 0.06 decrease in NLPD.

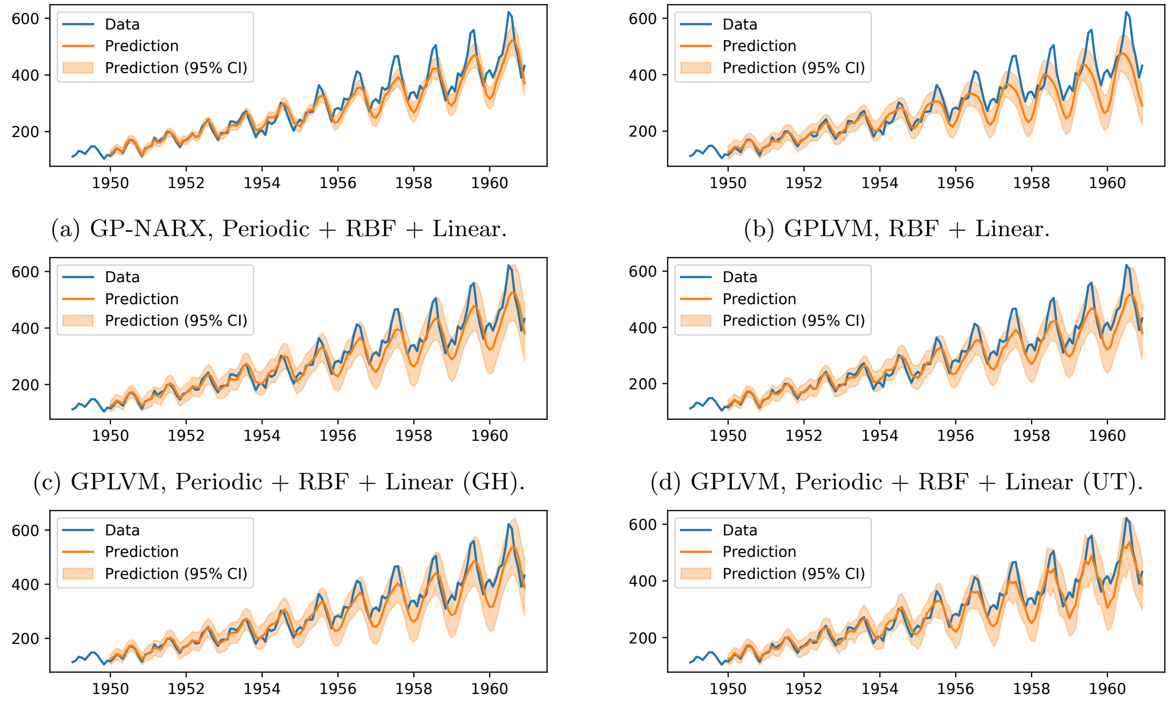
Method	# evaluations	Kernel	NLPD	RMSE
GP-NARX	-	RBF+Linear	11.37	69.40
	-	Per.+RBF+Lin.	7.46	44.98
GPLVM - Analytic	-	RBF+Linear	7.08	68.93
GPLVM - GH	4096	RBF+Linear	7.07	68.88
		Per.+RBF+Lin.	5.20	45.00
GPLVM - UT	24	RBF+Linear	7.10	69.11
		Per.+RBF+Lin.	5.26	45.27
GPLVM - MC	24	RBF+Linear	7.52 ± 0.41	71.16 ± 3.15
		Per.+RBF+Lin.	5.41 ± 0.17	46.99 ± 3.04
	200	RBF+Linear	7.09 ± 0.20	68.82 ± 2.09
		Per.+RBF+Lin.	5.19 ± 0.06	45.19 ± 1.32
	4096	RBF+Linear	7.07 ± 0.03	68.81 ± 0.37
		Per.+RBF+Lin.	5.19 ± 0.01	45.29 ± 0.28

5 RELATED WORK

A few authors have considered the UT in the context of GP models. For instance, Ko and Fox (2009) and Ko, Klein, et al. (2007) propose using the Unscented Kalman Filter (UKF) with GP-based transition and observation functions, and others have successfully applied the resulting GP-UKF (Anger et al. 2012; Safarinejadian and Kowsari 2014; Wang et al. 2014). Ko and Fox (2010) extend the previous works by considering the original GPLVM (Lawrence 2004), where the latent variables are optimized, instead of integrated. Steinberg and Bonilla (2014) tackle other kinds of intractabilities and use the UT in GP models with non-Gaussian likelihoods in a variational framework. The resulting Unscented GP (UGP) is evaluated in synthetic inversion problems and binary classification. Later, Bonilla et al. (2016) generalize that methodology to solve multi-output and multi-task problems while also enabling non-Gaussian likelihoods.

In summary, the GP-UKF and related models use GPs for filtering by basing their models on unscented Kalman filters. Furthermore, the UGP and related models focus on using the UT to solve intractable integrals that arise when considering non-Gaussian likelihoods in GP models. However, this paper's subject matter is the use of arbitrary kernels through the UT in Bayesian GPLVM models, where the latent variables that represent uncertain inputs are approximately marginalized.

[3] As defined by MacKay (1998) at Eq. (47).



(a) GP-NARX, Periodic + RBF + Linear.

(b) GPLVM, RBF + Linear.

(c) GPLVM, Periodic + RBF + Linear (GH).

(d) GPLVM, Periodic + RBF + Linear (UT).

(e) GPLVM, Periodic + RBF + Linear (MC(4096)).

(f) GPLVM, Periodic + RBF + Linear (MC(24)).

Figure 3: Results obtained in the dynamical free simulation experiments. Best obtained runs are shown. Visibly, the MC approximation with 24 points has a much lower quality in its mean compared to the UT approximation by failing to model the peaks of the curve properly.

6 CONCLUSION

This paper considers learning Bayesian GPLVM models using arbitrary kernels. More specifically, we use the UT to tackle the intractabilities that arise in the popular VI scheme by Titsias and Lawrence (2010).

We perform experiments on two tasks: dimensionality reduction and free simulation of dynamical models with uncertainty propagation. In both cases, the UT-based approach scales much better than the Gauss-Hermite quadrature while obtaining a similar overall approximation in our experiments. The UT results are also more stable and consistent than those obtained by Monte Carlo sampling, which may require a more significant number of samples and is not compatible with the popular quasi-Newton BFGS optimization algorithm. Notably, the method is simple to implement and maintains the deterministic nature of the standard VI for Bayesian GLPLVMs.

For future work, we aim to evaluate how other methods of obtaining sigma points might increase or decrease the quality of the approximations taken. Also, we intend to assess the UT in more scenarios where inference with GP models falls into intractable expectations. For instance, we could tackle integrals that arise from DGP models for which inference is intractable due to low-dimensional integrals, like the doubly stochastic Gaussian process by Salimbeni and Deisenroth (2017) and recurrent Gaussian processes by Mattos et al. (2016).

Acknowledgements

We gratefully acknowledge the support from National Council for Scientific and Technological Development (CNPq grant 302289/2019-4), Delfos Intelligent Maintenance, and Fundação ASTEF (FASTEF grant F0229).

References

Anger, Christoph, Robert Schrader, and Uwe Klingauf (2012). “Unscented Kalman filter with Gaussian process degradation model for bearing fault prognosis”. In: *Proceedings of the european conference of the PHM society*. Ed. by Anibal Bregon and Abhinav Saxena. Dresden, Germany.

Bishop, Christopher M. and Gavin D. James (1993). “Analysis of multiphase flows using dual-energy gamma densitometry and neural networks”. In: *Nuclear Instruments and Methods in Physics Research*

Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 327.2-3, pp. 580–593.

Bonilla, Edwin, Daniel Steinberg, and Alistair Reid (2016). "Extended and Unscented Kitchen Sinks". In: ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, pp. 1651–1659.

Byrd, Richard H. et al. (1995). "A limited memory algorithm for bound constrained optimization". In: *SIAM Journal on Scientific Computing* 16.5, pp. 1190–1208.

Calandra, Roberto et al. (2016). "Manifold Gaussian processes for regression". In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE. Vancouver, BC, Canada: IEEE, pp. 3338–3345.

Casale, Francesco Paolo et al. (2018). "Gaussian Process Prior Variational Autoencoders". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio et al. Curran Associates, Inc., pp. 10369–10380.

Damianou, Andreas and Neil Lawrence (2013). "Deep Gaussian processes". In: *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Carlos M. Carvalho and Pradeep Ravikumar. Vol. 31. Proceedings of Machine Learning Research. Scottsdale, AZ, USA: PMLR, pp. 207–215.

Damianou, Andreas C., Michalis K. Titsias, and Neil D. Lawrence (2016). "Variational inference for latent variables and uncertain inputs in Gaussian processes". In: *Journal of Machine Learning Research* 17.42, pp. 1–62.

Duvenaud, David et al. (2013). "Structure discovery in nonparametric regression through compositional kernel search". In: *Proceedings of the 30th International Conference on Machine Learning*. Ed. by Sanjoy Dasgupta and David McAllester. Vol. 28. Proceedings of Machine Learning Research 3. Atlanta, GA, USA: PMLR, pp. 1166–1174.

Eleftheriadis, Stefanos, Tom Nicholson, et al. (2017). "Identification of Gaussian process state space models". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., pp. 5309–5319.

Eleftheriadis, Stefanos, Ognjen Rudovic, et al. (2017). "Variational Gaussian Process Auto-Encoder for Ordinal Prediction of Facial Action Units". In: *Computer Vision - ACCV 2016*. Ed. by Shang-Hong Lai et al. Cham: Springer International Publishing, pp. 154–170.

G. Matthews, Alexander G. de et al. (2017). "GPflow: A Gaussian Process Library using TensorFlow". In: *Journal of Machine Learning Research* 18.40, pp. 1–6.

Girard, Agathe et al. (2003). "Gaussian process priors with uncertain inputs application to multiple-step ahead time series forecasting". In: *Advances in Neural Information Processing Systems 15*. Ed. by S. Becker, S. Thrun, and K. Obermayer. MIT Press, pp. 545–552.

Havasi, Marton, Jose Miguel Hernandez-Lobato, and Juan José Murillo-Fuentes (2018). "Inference in Deep Gaussian Processes using Stochastic Gradient Hamiltonian Monte Carlo". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio et al. Curran Associates, Inc., pp. 7506–7516.

Hyndman, Rob J (2018). *Time Series Data Library.*

Jordan, Michael I. et al. (1999). "An introduction to variational methods for graphical models". In: *Machine Learning* 37.2, pp. 183–233.

Julier, S.J. and J.K. Uhlmann (2004). "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE* 92.3, pp. 401–422.

Kingma, Diederik P. and Max Welling (2014). "Auto-Encoding Variational Bayes". In: *2nd International Conference on Learning Representations, ICLR 2014*. Ed. by Yoshua Bengio and Yann LeCun. Banff, AB, Canada.

Ko, Jonathan and Dieter Fox (2009). "GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models". In: *Autonomous Robots* 27.1, pp. 75–90.

– (2010). "Learning GP-BayesFilters via Gaussian process latent variable models". In: *Autonomous Robots* 30.1, pp. 3–23.

Ko, Jonathan, Daniel J. Klein, et al. (2007). "GP-UKF: Unscented Kalman filters with Gaussian process prediction and observation models". In: *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. IEEE, pp. 1901–1907.

Kocijan, Juš et al. (2005). "Dynamic systems identification with Gaussian processes". In: *Mathematical and Computer Modelling of Dynamical Systems* 11.4, pp. 411–424.

Lawrence, Neil D. (2004). "Gaussian Process Latent Variable Models for Visualisation of High Dimensional Data". In: *Advances in Neural Information Processing Systems 16*. Ed. by S. Thrun, L. K. Saul, and B. Schölkopf. MIT Press, pp. 329–336.

Lloyd, James Robert et al. (2014). "Automatic Construction and Natural-Language Description of Nonparametric Regression Models". In: *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*. AAAI'14. Québec City, Québec, Canada: AAAI Press, pp. 1242–1250.

MacKay, D. J. C. (1998). "Introduction to Gaussian Processes". In: *Neural Networks and Machine*

Learning. Ed. by C. M. Bishop. NATO ASI Series. Kluwer Academic Press, pp. 133–166.

Mattos, César Lincoln C. et al. (2016). "Recurrent Gaussian Processes". In: *4th International Conference on Learning Representations, ICLR 2016*. Ed. by Yoshua Bengio and Yann LeCun.

Menegaz, Henrique M. T. et al. (2015). "A Systematization of the Unscented Kalman Filter Theory". In: *IEEE Transactions on Automatic Control* 60.10, pp. 2583–2598.

Rasmussen, Carl and Chris Williams (2006). *Gaussian Processes for Machine Learning*. 1st ed. Cambridge, MA, USA: MIT Press.

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra (2014). "Stochastic Backpropagation and Approximate Inference in Deep Generative Models". In: *Proceedings of the 31st International Conference on Machine Learning*. Ed. by Eric P. Xing and Tony Jebara. Vol. 32. Proceedings of Machine Learning Research 2. Bejing, China: PMLR, pp. 1278–1286.

Safarinejadian, Behrooz and Elham Kowsari (2014). "Fault detection in non-linear systems based on GP-EKF and GP-UKF algorithms". In: *Systems Science & Control Engineering* 2.1, pp. 610–620.

Salimbeni, Hugh and Marc Deisenroth (2017). "Doubly Stochastic Variational Inference for Deep Gaussian Processes". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., pp. 4588–4599.

Al-Shedivat, Maruan et al. (2017). "Learning scalable deep kernels with recurrent structure". In: *Journal of Machine Learning Research* 18.82. Ed. by Kevin Murphy and Bernhard Schölkopf, pp. 1–37.

Steinberg, Daniel M and Edwin V Bonilla (2014). "Extended and Unscented Gaussian Processes". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., pp. 1251–1259.

Titsias, Michalis K. (2009). "Variational Learning of Inducing Variables in Sparse Gaussian Processes". In: *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*. Ed. by David van Dyk and Max Welling. Vol. 5. Proceedings of Machine Learning Research. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR, pp. 567–574.

Titsias, Michalis K. and Neil D. Lawrence (2010). "Bayesian Gaussian Process Latent Variable Model". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Yee Whye Teh and Mike Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, pp. 844–851.

Titsias, Michalis K. and Miguel Lázaro-Gredilla (2014). "Doubly Stochastic Variational Bayes for Non-Conjugate Inference". In: *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*. ICML'14. Beijing, China: JMLR.org, pp. II–1971–II–1980.

Uhlmann, Jeffrey K. (1995). "Dynamic map building and localization: New theoretical foundations". PhD thesis. University of Oxford.

Wang, Ziyou et al. (2014). "A human motion estimation method based on GP-UKF". In: *2014 IEEE International Conference on Information and Automation (ICIA)*. IEEE. Hailar, China: IEEE, pp. 1228–1232.

Wilson, Andrew and Ryan Adams (2013). "Gaussian Process Kernels for Pattern Discovery and Extrapolation". In: *Proceedings of the 30th International Conference on Machine Learning*. Ed. by Sanjoy Dasgupta and David McAllester. Vol. 28. Proceedings of Machine Learning Research 3. Atlanta, GA, USA: PMLR, pp. 1067–1075.

Wilson, Andrew G, Zhiting Hu, et al. (2016a). "Stochastic Variational Deep Kernel Learning". In: *Advances in Neural Information Processing Systems 29*. Ed. by D. D. Lee et al. Curran Associates, Inc., pp. 2586–2594.

Wilson, Andrew Gordon, Zhiting Hu, et al. (2016b). "Deep Kernel Learning". In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. Ed. by Arthur Gretton and Christian C. Robert. Vol. 51. Proceedings of Machine Learning Research. Cadiz, Spain: PMLR, pp. 370–378.