Residual Deep Gaussian Processes on Manifolds for Geometry-aware Bayesian Optimization on Hyperspheres

Kacper Wyrwal WYRWAL.KACPER@GMAIL.COM
University of Edinburgh; ETH Zürich
Viacheslav Borovitskiy VIACHESLAV.BOROVITSKIY@GMAIL.COM
ETH Zürich

Abstract

Gaussian processes (GPs) are a widely-used model class for approximating unknown functions, especially useful in tasks such as Bayesian optimisation, where accurate uncertainty estimates are key. Deep Gaussian processes (DGPs) are a multi-layered generalisation of GPs, which promises improved performance at modelling complex functions. Some of the problems where GPs and DGPs may be utilised involve data on manifolds like hyperspheres. Recent work has recognised this, generalising scalar-valued and vector-valued Matérn GPs to a broad class of Riemannian manifolds. Despite that, an appropriate analogue of DGP for Riemannian manifolds is missing. We introduce a new model, *residual manifold DGP*, and a suitable doubly stochastic variational inference technique that helps train and deploy it on hyperspheres. Through examination on stylised examples, we highlight the usefulness of residual deep manifold GPs on regression tasks and in Bayesian optimisation.

Keywords: Deep Gaussian processes, Riemannian manifolds, Bayesian optimisation

1. Introduction

Gaussian processes (GPs) are a powerful probabilistic approach for modelling unknown functions. Owing to their accurate uncertainty estimates, GPs have found wide-spread success in tasks such as Bayesian optimisation (Shahriari et al., 2016), active learning (Krause et al., 2008), and reinforcement learning (Rasmussen and Kuss, 2004). Inevitably, in some practical problems the unknown function is complex and non-smooth. Single-layer GPs can struggle to model such functions due to the simplicity bias of the commonly used kernels. One way to tackle this limitation is to sequentially compose multiple GPs forming a deep GP. This layered structure has been shown to improve performance on a variety of tasks (Damianou and Lawrence, 2013; Salimbeni and Deisenroth, 2017).

Recent work has introduced an exciting frontier to Gaussian process research, namely Gaussian processes on Riemannian manifolds. This is especially important because in many areas of interest, such as robotics and climate science, data is inherent to non-Euclidean manifolds. The Matérn kernel has been generalised to a wide class of manifolds (Borovitskiy et al., 2020; Azangulov et al., 2023a,b), allowing for construction of scalar-valued and vector-valued manifold GPs (Hutchinson et al., 2021). However, an appropriate analogue of deep GPs in the manifold setting is missing.

. Code available at: `https://github.com/KacperWyrwal/residual-manifold-deep-gp`

In this paper we introduce a generalisation of deep GPs to Riemannian manifolds. In our construction, each hidden layer is a manifold-to-manifold map that models a difference from the identity map, a *residual*. Because each layer only learns a residual function with respect to the inputs, we call our model the residual manifold deep GP. We implement residual manifold deep GPs on hyperspheres and evaluate them on synthetic experiments including Bayesian optimisation. Although we focus on hyperspheres, it should be possible to extend these models to any manifold where a Gaussian vector field—see Section 3—is implemented. We find that our model can improve over the shallow manifold GPs for modelling complex functions in the larger data regime.

2. Background

In this section, we detail the necessary background on *DGPs* and *manifold Matérn Gaussian processes*. We also revise the *doubly stochastic variational inference* technique for DGPs (Salimbeni and Deisenroth, 2017).

Deep Gaussian process Deep GPs are a multi-layered generalisation of GPs. In contrast to single-layer GPs, inference in deep GPs is intractable and requires approximations. A widely-used and successful framework for such approximations is doubly stochastic variational inference. In this framework, the intractable posterior is approximated by a deep GP whose layers are vector-valued GPs with independent *sparse GP* components.

A sparse GP f is a Gaussian process posterior conditioned by the *inducing distribution* q at a small set of *inducing locations* $\mathbf{Z}$. A posterior under a DGP prior is approximated by a composition of sparse Gaussian processes

$$f_L \circ \cdots \circ f_1 \tag{1}$$

each with its own $q_i \sim N(\boldsymbol{\mu}_i, \mathbf{K}_i)$ and $\mathbf{Z}_i$. Variational inference finds $\boldsymbol{\mu}_i$, $\mathbf{K}_i$ and $\mathbf{Z}_i$ by minimising the KL divergence between the process in Equation (1) and the true posterior.

Computing expectations and variances of the approximation in Equation (1) is still intractable; however, it can be approximated with Monte Carlo sampling. Because deep GPs are a simple composition of sparse GPs, this can be done by layer-wise sampling.

Salimbeni and Deisenroth (2017) modify DGPs taking inspiration from the skip connection of the ResNet model. Specifically, when the input and output dimensions of a hidden layer are equal, the inputs are added to the output mean. We will show in Section 3 that our model is a strict manifold generalisation of this construction.

Efficient sampling for variational deep GP posteriors To approximate expectations and variances of the variational deep GP posterior we need to draw samples from it. Doing this naively for n sampling locations would take $O(n^3)$ time. However, *pathwise conditioning* Wilson et al. (2020, 2021) allows us to sample approximately in $O(n)$ time. This method applies if a weight-space approximation of the kernel exists. We will see in the following paragraph that this is true for Matérn kernels on compact manifolds[1].

1. For non-compact *symmetric spaces*, like hyperbolic spaces or manifolds of positive definite matrices, an appropriate approximation exists as well Azangulov et al. (2023b). It is akin to *random Fourier features*.

Manifold Matérn kernel Borovitskiy et al. (2020) show that on a compact Riemannian manifold X a Matérn kernel $k : X \times X \to \mathbb{R}$ with length scale κ and smoothness ν and average variance[2] σ^2 may be expressed as

$$k_\nu(\boldsymbol{x}, \boldsymbol{x'}) = \frac{\sigma^2}{C_\nu} \sum_{n=0}^{\infty} a_{\nu,\kappa}(\lambda_n) f_n(\boldsymbol{x}) f_n(\boldsymbol{x'}), \quad a_{\nu,\kappa}(\lambda_n) = \begin{cases} \left(\frac{2\nu}{\kappa^2} + \lambda_n\right)^{-\nu - \frac{d}{2}}, & \text{for } \nu \in (0, \infty), \\ e^{-\frac{\kappa^2}{2}\lambda_n}, & \text{for } \nu = \infty. \end{cases} \tag{2}$$

Here λ_n, f_n are the eigenpairs of the Laplace–Beltrami operator on X and C_ν is a constant. One can sensibly approximate k by truncating the infinite sum in Equation 2 to the first N terms (Rosa et al., 2023). This can be expressed as a weight-space approximation

$$k(\boldsymbol{x}, \boldsymbol{x'}) \approx \sum_{n=1}^{N} a_{\nu,\kappa}(\lambda_n) f_n(\boldsymbol{x}) f_n(\boldsymbol{x'}) = \boldsymbol{\Phi}(\boldsymbol{x}) \boldsymbol{\Phi}(\boldsymbol{x})^\mathsf{T}. \tag{3}$$

with $\boldsymbol{\Phi}(\boldsymbol{x}) = \left[\sqrt{a_{\nu,\kappa}(\lambda_1)} f_1(\boldsymbol{x}), \cdots, \sqrt{a_{\nu,\kappa}(\lambda_N)} f_N(\boldsymbol{x})\right]$. Thus, pathwise conditioning applies.

3. Model Construction

In this section we present the residual manifold deep GP. The key challenge we address is constructing manifold-to-manifold GPs for the hidden layers. This problem in general is far from trivial. On one hand, designing manifold-input GPs is a whole research direction. On the other hand, manifold-output GPs cannot be Gaussian in the usual sense, since Gaussian distributions are vector-valued Mallasto and Feragen (2018). We work around this difficulty by modelling displacement vectors (the residuals) which give the manifold-to-manifold maps when composed with the exponential map on the manifold. The problem is thus shifted towards modelling Gaussian vector fields on manifolds.

Gaussian vector fields We discuss two practical methods for constructing Gaussian vector fields on a d-dimensional Riemannian manifold X. First, Hutchinson et al. (2021) embed X in $\mathbb{R}^n$ with an embedding emb and defines a Euclidean vector-valued GP $\boldsymbol{f}$ on $\mathrm{emb}(X)$. The output of $\boldsymbol{f}$ is then projected into the tangent bundle $\mathcal{T}X$ with a position-dependent linear projection $\mathbf{P}_{\boldsymbol{x}}$. A Gaussian vector field $\boldsymbol{f'}$ on X is thus given by

$$\boldsymbol{f'}(\boldsymbol{x}) = \mathbf{P}_{\mathrm{emb}(\boldsymbol{x})} \boldsymbol{f}(\mathrm{emb}(\boldsymbol{x})). \tag{4}$$

Second, a natural method is to define vector fields in terms of a coordinate frame on X—that is, a set of functions $\{e_i\}_{i=1}^d : X \to \mathcal{T}X$ such that $\{e_i(\boldsymbol{x})\}_{i=1}^d$ spans $\mathcal{T}_{\boldsymbol{x}}X$ for every $\boldsymbol{x} \in X$. Taking d independent scalar-valued GPs $\{f_i\}_{i=1}^d$ that act as coefficients for the coordinate frame, a Gaussian vector field $\boldsymbol{f'}$ on X is given by

$$\boldsymbol{f}(\boldsymbol{x}) = f_1(\boldsymbol{x}) e_1(\boldsymbol{x}) + ... + f_d(\boldsymbol{x}) e_d(\boldsymbol{x}). \tag{5}$$

All Gaussian vector fields can be obtained by either construction; however, the choice of a coordinate frame is not obvious a priori and, as seen in Section 4, has significant influence on model performance.

2. The value of $k(\boldsymbol{x}, \boldsymbol{x})$ need not be equal for every $\boldsymbol{x} \in X$. This is because k is only invariant under the action of the manifold symmetry group which can fail to act *transitively* on X.

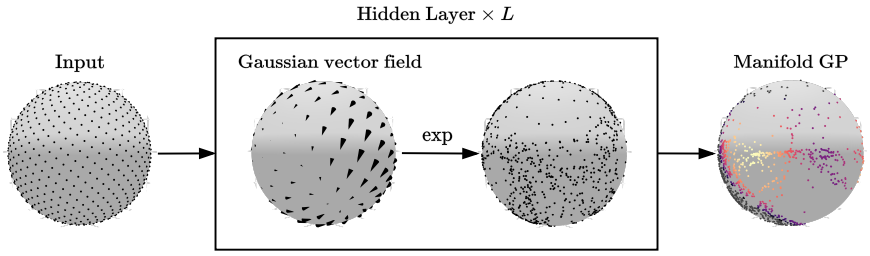

Figure 1: Schematic illustration of the residual manifold deep Gaussian Process with L hidden layers. Hidden layers are a composition of Gaussian vector fields with the exponential map. Output layer is a manifold GP - in this case scalar valued.

Residual manifold deep GP construction With a way of building Gaussian vector fields, we can build a hidden layer of the residual manifold deep GP by composing Gaussian vector field with the *exponential map* which "projects" tangent vectors onto the manifold. The output layer is a manifold GP chosen appropriately to the given task. We illustrate the construction of residual manifold deep GPs schematically in Figure 1. Remarkably, because we can use sparse manifold GPs to build Gaussian vector fields, our model enjoys efficient approximate training and inference via doubly stochastic variational inference, as well as a linear time approximate posterior computation via Monte Carlo sampling as in Section 2.

In fact, it turns out that our model generalises the deep GP presented in the doubly stochastic variational inference framework. Indeed, on the Euclidean manifold $X = \mathbb{R}^n$ we can identity the tangent space $\mathcal{T}_{\boldsymbol{x}}\mathbb{R}^n$ with $\mathbb{R}^n$ and exponential map with vector addition $\exp_{\boldsymbol{x}}(\boldsymbol{x'}) = \boldsymbol{x} + \boldsymbol{x'}$. As we saw in Section 2, this is exactly the model presented by Salimbeni and Deisenroth (2017) when the input and output dimension of each hidden layer is equal.

4. Experiments

Our driving practical goal of developing a deep GP on manifolds was to improve upon the performance of shallow manifold GPs on modelling complex, non-smooth functions. In this section, we test whether the residual manifold deep GP achieves this goal.

To this end, we implement the residual manifold deep GP on hyperspheres and perform a set of experiments on synthetic data. First, we benchmark our model on a regression task with a complex, non-smooth target function, examining the effect of data density and model depth on performance. Then, we focus on the coordinate frame variant of our model, highlighting the benefits of a well-chosen coordinate frame and showcasing an example of a learnable coordinate frame. Finally, we demonstrate the potential of residual manifold deep GPs for Bayesian optimisation on a stylised example.

Model depth and data density We test our model on a regression task with a custom non-smooth ground truth function shown in Figure 3 (right). We use the projected implementation of Gaussian vector fields, to avoid the choice of a coordinate frame which can have a significant impact on performance. We examine model performance across 0-4

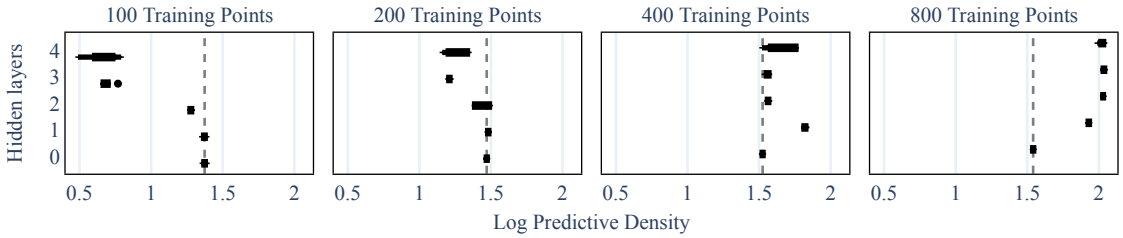

Figure 2: Log predictive density of residual manifold deep GPs on a regression task with the target function shown in Figure 3. Dotted line indicates mean log predictive density of a sparse manifold GP. Boxes show the interquartile range.

hidden layers and 100, 200, 400, and 800 training points. For each configuration of model depth and data regime we train the model for 1000 steps and then measure its log predictive density on 2000 points, presenting the results in Figure 2. We repeat each train-test run 5 times. Each sparse GP in every model configuration uses 60 inducing points. Training, testing, and inducing points are all arranged in an approximately uniform grid on $\mathbb{S}^2$.

We find that our model outperforms shallow sparse GPs in larger data regimes, while the opposite is true in small data regimes. This is not surprising and aligns with the results of Salimbeni and Deisenroth (2017) in the Euclidean case. Nevertheless, we may notice that only the model with a single hidden layer is never worse than the shallow GP. Thus, although, in principle, residual manifold deep GPs can always recover the shallow solution, this may not always happen in practice.

Coordinate frames Theoretically, every Gaussian vector field can be obtained with any coordinate frame (Hutchinson et al., 2021). Practically, however, we find that the choice of a coordinate frame has significant impact on the performance of our model. Figure 3 showcases this on on residual manifold deep GP with 1 hidden layer. Each model in the plot was trained for 1000 steps, yet only the well-chosen coordinate frame gives an almost perfect fit. To avoid a poor choice of coordinate we tried a parametrised approach optimised jointly with the model. This already yields a much better fit than the model with a poorly chosen frame, suggesting that learnable coordinate frames may be worth further investigation.

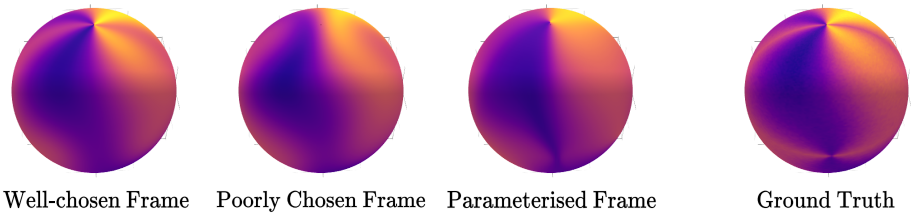

Figure 3: Posterior means of residual manifold deep GPs with one hidden layer across different coordinate frames used to construct Gaussian vector fields.

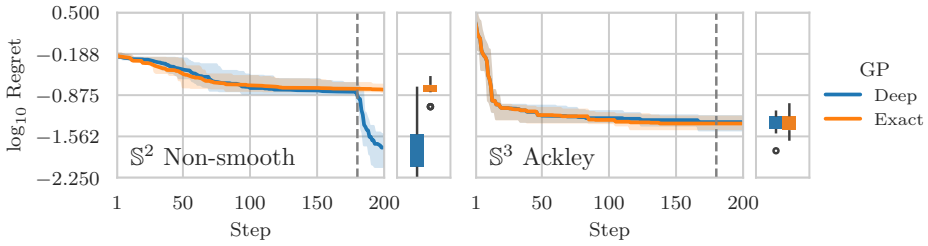

Figure 4: Logarithm of the regret for a function with a non-smooth minimum on $\mathbb{S}^2$ (left) and a smooth function with a smooth minimum on $\mathbb{S}^3$ (right). Dotted line indicates the transition to a residual manifold deep GP. Solid lines indicate the mean regret. Boxes and shaded regions show the interquartile range.

Bayesian optimisation We examine the potential of our model in Bayesian optimisation on the Ackley function projected onto $\mathbb{S}^3$—following Jaquier et al. (2021)—and on a custom non-smooth function on $\mathbb{S}^2$ in Figure 3 (right). Based on insight from the previous experiments we tested a two-hidden-layer model on the Ackley function and a one-hidden-layer coordinate frame variant of our model on the custom function.

Our previous experiments have shown that our model requires sufficiently dense data to work well. Thus, we perform the first 180 acquisition steps with an exact GP before switching to our deep GP. We compare the log regret of the process to the baseline obtained from 200 iterations with an exact GP. Uncertainty estimates in Figure 4 are obtained by repeating each optimisation process 10 times.

We find that, with the Ackley, function switching to our model offers no advantage. This is not surprising, since the minimum of the Ackley function is located in a smooth landscape. However, for the custom non-smooth function our model makes a significant and often immediate improvement. One strategy springing from this intuition would be to switch between a shallow and deep models according to some schedule.

5. Conclusion

We have utilised a range of recent advancements in manifold GP to construct the residual manifold deep GP, a deep GP composed of manifold-to-manifold maps, and provided the computational techniques needed to use it. Its layers, manifold-to-manifold GPs, are Gaussian vector fields composed with the exponential map. Each of them models the displacement of its inputs, i.e. the residual difference from the identity.

We have implemented residual manifold deep GPs on hyperspheres and examined its performance on different stylised tasks, demonstrating that it consistently outperforms sparse manifold GPs in regression when the target function is complex and data is sufficiently dense. Finally, we have shown that residual manifold deep GPs can benefit geometry-aware Bayesian optimisation when the landscape around the minimum is complex. Future research may explore ways of using exact manifold GPs and residual manifold deep GPs in tandem for a best-of-both-worlds approach.

References

Iskander Azangulov, Andrei Smolensky, Alexander Terenin, and Viacheslav Borovitskiy. Stationary kernels and gaussian processes on lie groups and their homogeneous spaces i: the compact case, 2023a.

Iskander Azangulov, Andrei Smolensky, Alexander Terenin, and Viacheslav Borovitskiy. Stationary kernels and gaussian processes on lie groups and their homogeneous spaces ii: non-compact symmetric spaces, 2023b.

Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Matérn gaussian processes on riemannian manifolds, 2020.

Andreas C. Damianou and Neil D. Lawrence. Deep gaussian processes, 2013.

Michael Hutchinson, Alexander Terenin, Viacheslav Borovitskiy, So Takao, Yee Whye Teh, and Marc Peter Deisenroth. Vector-valued gaussian processes on riemannian manifolds via gauge independent projected kernels, 2021.

Noémie Jaquier, Viacheslav Borovitskiy, Andrei Smolensky, Alexander Terenin, Tamim Asfour, and Leonel Rozo. Geometry-aware bayesian optimization in robotics using riemannian matérn kernels, 2021.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, 2008.

Anton Mallasto and Aasa Feragen. Wrapped gaussian process regression on riemannian manifolds. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2018.

Carl Edward Rasmussen and Malte Kuss. Gaussian processes in reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 751–758, 2004.

Paul Rosa, Viacheslav Borovitskiy, Alexander Terenin, and Judith Rousseau. Posterior contraction rates for matérn gaussian processes on riemannian manifolds, 2023.

Hugh Salimbeni and Marc Deisenroth. Doubly stochastic variational inference for deep gaussian processes, 2017.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

James Townsend, Niklas Koep, and Sebastian Weichwald. Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation. *Journal of Machine Learning Research*, 17(137):1–5, 2016.

James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Efficiently sampling functions from gaussian process posteriors, 2020.

James T Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Pathwise conditioning of gaussian processes. *Journal of Machine Learning Research*, 22(105):1–47, 2021.

Appendix A. Additional experimental details

Coordinate frames The well-chosen coordinate frame we use is defined at $\boldsymbol{x} = (x, y, z) = (\sin(\phi)\cos(\theta), \sin(\phi)\sin(\theta), \cos(\phi)) \in \mathbb{S}^2$ by

$$(e_1(\boldsymbol{x}), e_2(\boldsymbol{x})) = \begin{cases} (\partial_\theta \boldsymbol{x}, \partial_\phi \boldsymbol{x}) & \text{for } (\phi, \theta) \in (0, \pi) \times [0, 2\pi) \\ ((0, 1, \pm 1), (1, 0, \pm 1)) & \text{for } z = \pm 1 \end{cases}. \tag{6}$$

It turns out to be particularly useful for modelling the ground truth function in Figure 3 because they both have singularity points at $(0, 0, 1)$ and $(0, 0, -1)$. We obtain the poor choice of coordinate frame simply by rotating the well-chosen coordinate frame. Specifically, we orient it such that its singularity points are located in a region where the target function smooth.

For the coordinate frame parameterisation we used $\mathbf{A} \in \mathbb{R}^{3\times 3}$ and $\boldsymbol{b} \in \mathbb{R}^3$. We use $\mathbf{A}$ and $\boldsymbol{b}$ to construct a coordinate frame (e_1, e_2) on the sphere with the following formula

$$e_1(\boldsymbol{x}) = \frac{(\mathbf{A}\boldsymbol{x} + \boldsymbol{b}) \times \boldsymbol{x}}{\|(\mathbf{A}\boldsymbol{x} + \boldsymbol{b}) \times \boldsymbol{x}\|}, \qquad e_2(\boldsymbol{x}) = \frac{e_1(\boldsymbol{x}) \times \boldsymbol{x}}{\|e_1(\boldsymbol{x}) \times \boldsymbol{x}\|}. \tag{7}$$

Bayesian optimisation Drawing a connection to (Jaquier et al., 2021), we conduct a 200-step Bayesian optimisation process with 5 random initial observations. We use the expected improvement acquisition function and find its minimum using a geometry-aware method implemented in Pymanopt (Townsend et al., 2016). In contrast to Jaquier et al. (2021) we use a first-order method instead of a second-order method, which seems not to result in a visible difference in performance. We use the expected improvement acquisition function and after each acquisition step we retrain the model from scratch for 500 iterations.