Spherical Inducing Features for Orthogonally-Decoupled Gaussian Processes

Louis C. Tiao [1*]   Vincent Dutordoir [2 3]   Victor Picheny [3]

Abstract

Despite their many desirable properties, Gaussian processes (GPs) are often compared unfavorably to deep neural networks (NNs) for lacking the ability to learn representations. Recent efforts to bridge the gap between GPs and deep NNs have yielded a new class of inter-domain variational GPs in which the inducing variables correspond to hidden units of a feedforward NN. In this work, we examine some practical issues associated with this approach and propose an extension that leverages the orthogonal decomposition of GPs to mitigate these limitations. In particular, we introduce spherical inter-domain features to construct more flexible data-dependent basis functions for both the principal and orthogonal components of the GP approximation and show that incorporating NN activation features under this framework not only alleviates these shortcomings but is more scalable than alternative strategies. Experiments on multiple benchmark datasets demonstrate the effectiveness of our approach.

1. Introduction

Gaussian processes (GPs) provide a powerful framework for reasoning about unknown functions and are ubiquitous in probabilistic machine learning (Rasmussen & Williams, 2006). They are data-efficient, innately robust to over-fitting, and can flexibly encode priors through their covariance function. Last but not least, by virtue of their ability to faithfully capture predictive uncertainty, they form the backbone of many sequential decision-making methods that require reliable uncertainty estimates to balance trade-offs such as exploration and exploitation, e.g. in reinforcement learning (Deisenroth & Rasmussen, 2011), Bayesian optimization (Garnett, 2023), and probabilistic numerics (Hennig

et al., 2022). In spite of their many advantages, GPs are often compared unfavourably to deep neural networks (NNs) for their poor scalability to large datasets, and their inability to capture rich hierarchies of abstract representations (Calandra et al., 2016; Wilson et al., 2016; Ober et al., 2021). While GPs are the infinite-width limit of NNs and therefore, in theory, have infinitely more basis functions (Neal, 1996), these basis functions are static and fully determined by the covariance function (MacKay et al., 1998). This makes it difficult for GPs to flexibly adapt to complex and structured data from which it is beneficial for the basis functions to learn and encode useful representations.

Considerable research effort has been devoted to sparse approximations for GPs (Csató & Opper, 2002; Seeger et al., 2003; Quinonero-Candela & Rasmussen, 2005; Snelson & Ghahramani, 2005). Not least of these is sparse variational GPs (SVGPs) (Titsias, 2009; Hensman et al., 2013; 2015a). Such advances have not only improved the scalability of GPs, but also unlocked more flexibility in model specification. In particular, the use of *inter-domain* inducing variables in SVGP (Lázaro-Gredilla & Figueiras-Vidal, 2009) effectively equips the GP approximation with data-dependent basis functions. Recent works have exploited this to construct a new family of SVGP models in which the basis functions correspond to activations of a feed-forward NN (Sun et al., 2020; Dutordoir et al., 2021). By stacking multiple layers to form a deep GP (DGP) (Damianou & Lawrence, 2013), the propagation of the predictive distribution accurately resembles a forward-pass through a deep NN.

In this paper, we show that while this approach results in a posterior predictive with a more expressive mean, the variance estimate is typically less accurate and tends to be over-dispersed. Additionally, we examine some practical challenges associated with this method, such as limitations on the use of certain popular kernel and NN activation choices. To address these issues, we propose an extension that aims to mitigate these limitations. Specifically, when viewed from the function-space perspective, the posterior predictive of SVGP depends on a single set of basis functions that is determined by only a finite collection of inducing variables. Recent advances introduce an orthogonal set of basis functions as a means of capturing additional variations remaining from the standard basis (Salimbeni et al., 2018; Cheng & Boots, 2017; Shi et al., 2020). We extend

this framework by introducing inter-domain variables to construct more flexible data-dependent basis functions for both the standard and orthogonal components. In particular, we show that incorporating NN activation inducing functions under this framework is an effective way to ameliorate the aforementioned shortcomings. Our experiments on numerous benchmark datasets demonstrate that this extension leads to improvements in predictive performance against comparable alternatives.

2. Background

Gaussian processes (GPs) are a flexible class of distributions over functions. A random function $f : \mathcal{X} \to \mathbb{R}$ on some domain $\mathcal{X} \subseteq \mathbb{R}^d$ is distributed according to a GP if, at any finite collection of input locations $\mathbf{X}_* \subseteq \mathcal{X}$, its values $\mathbf{f}_* = f(\mathbf{X}_*)$ follow a Gaussian distribution. A GP is fully determined by its mean function, which can be assumed without loss of generality to be constant (e.g. zero), and its covariance function $k(\mathbf{x}, \mathbf{x}')$.

Consider a supervised learning problem in which we have a dataset $\{\mathbf{x}_n, y_n\}_{n=1}^N$ consisting of scalar outputs y_n, which are related to $f_n \triangleq f(\mathbf{x}_n)$, the value of some unknown function $f(\cdot)$ at input $\mathbf{x}_n \in \mathcal{X}$, through the likelihood $p(y_n \mid f_n, \theta)$. A powerful modelling approach consists of specifying a GP prior on the latent function $f(\cdot)$,

$$f(\mathbf{x}) \sim \mathcal{GP}\left(0, k(\mathbf{x}, \mathbf{x}')\right). \qquad (1)$$

Let $\mathbf{X}$ denote the inputs, $\mathbf{f}$ the corresponding latent function values, and $\mathbf{y}$ the outputs. In the regression setting, the outputs $\mathbf{y}$ are noisy observations of the latent values $\mathbf{f}$, typically related through a Gaussian likelihood $p(\mathbf{y} \mid \mathbf{f}, \theta) = \mathcal{N}(\mathbf{y} \mid \mathbf{f}, \beta^{-1}\mathbf{I})$ for some precision $\beta > 0$. In this case, although the posterior $p(\mathbf{f} \mid \mathbf{y})$ is analytically tractable, its computation has time complexity $\mathcal{O}(N^3)$.

2.1. Sparse Gaussian processes

A range of sparse GP methods have been developed over the years to mitigate these limitations (Csató & Opper, 2002; Seeger et al., 2003; Quinonero-Candela & Rasmussen, 2005; Snelson & Ghahramani, 2005). Broadly, in sparse GPs, one summarizes $f(\cdot)$ succinctly in terms of *inducing variables*, which are values $\mathbf{u} \triangleq f(\mathbf{Z})$ taken at a collection of M (usually $M \ll N$) locations $\mathbf{Z} = [\mathbf{z}_1 \cdots \mathbf{z}_M]^\top$, where $\mathbf{z}_m \in \mathcal{X}$. Not least among these approaches is sparse variational GP (SVGP), which casts sparse GPs within the framework of variational inference (VI) (Titsias, 2009; Hensman et al., 2013; 2015a).

Specifically, the joint distribution of the model augmented by inducing variables $\mathbf{u}$ is $p(\mathbf{y}, \mathbf{f}, \mathbf{u}) = p(\mathbf{y} \mid \mathbf{f})p(\mathbf{f}, \mathbf{u})$ where $p(\mathbf{f}, \mathbf{u}) = p(\mathbf{f} \mid \mathbf{u})p(\mathbf{u})$ for prior $p(\mathbf{u}) = \mathcal{N}(\mathbf{0}, \mathbf{K_{uu}})$

and conditional

$$p(\mathbf{f} \mid \mathbf{u}) = \mathcal{N}(\mathbf{f} \mid \mathbf{Q_{fu}}\mathbf{u}, \mathbf{K_{ff}} - \mathbf{Q_{ff}}), \qquad (2)$$

where $\mathbf{Q_{ff}} \triangleq \mathbf{Q_{fu}}\mathbf{K_{uu}}\mathbf{Q_{uf}}$ and $\mathbf{Q_{fu}} \triangleq \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}$. The joint variational distribution is defined as $q(\mathbf{f}, \mathbf{u}) \triangleq p(\mathbf{f} \mid \mathbf{u})q(\mathbf{u})$, where $q(\mathbf{u}) \triangleq \mathcal{N}(\mathbf{m_u}, \mathbf{C_u})$ for variational parameters $\mathbf{m_u} \in \mathbb{R}^M$ and $\mathbf{C_u} \in \mathbb{R}^{M \times M}$ s.t. $\mathbf{C_u} \succeq 0$. Integrating out $\mathbf{u}$ yields the posterior predictive

$$q(\mathbf{f}_*) = \mathcal{N}\left(\mathbf{Q_{*u}}\mathbf{m_u}, \mathbf{K_{**}} - \mathbf{Q_{*u}}(\mathbf{K_{uu}} - \mathbf{C_u})\mathbf{Q_{u*}}\right),$$

where parameters $\mathbf{m_u}$ and $\mathbf{C_u}$ are learned by minimizing the Kullback–Leibler (KL) divergence between the approximate and exact posterior, $\text{KL}\left[q(\mathbf{f}) \parallel p(\mathbf{f} \mid \mathbf{y})\right]$. Thus seen, SVGP has time complexity $\mathcal{O}(M^3)$ at prediction time and $\mathcal{O}(M^3 + M^2 N)$ during training.

In the reproducing kernel Hilbert space (RKHS) associated with k, the predictive has a dual representation in which the mean and covariance share the same basis determined by $\mathbf{u}$ (Cheng & Boots, 2017; Salimbeni et al., 2018). More specifically, the basis function is effectively the vector-valued function $\mathbf{k_u} : \mathcal{X} \to \mathbb{R}^M$ whose m-th component is defined as

$$[\mathbf{k_u}(\mathbf{x})]_m \triangleq \text{Cov}\left(f(\mathbf{x}), u_m\right). \qquad (3)$$

In the standard definition of inducing points, $[\mathbf{k_u}(\mathbf{x})]_m = k(\mathbf{z}_m, \mathbf{x})$, so the basis function is solely determined by k and the local influence of pseudo-input $\mathbf{z}_m$.

Inter-domain inducing features are a generalization of standard inducing variables in which each variable $u_m \triangleq L_m[f]$ for some linear operator $L_m : \mathbb{R}^{\mathcal{X}} \to \mathbb{R}$. A particularly useful operator is the integral transform, $L_m[f] \triangleq \int_{\mathcal{X}} f(\mathbf{x})\phi_m(\mathbf{x})\, d\mathbf{x}$, which was originally employed by Lázaro-Gredilla & Figueiras-Vidal (2009). Refer to the manuscript of van der Wilk et al. (2020) for a more thorough and contemporary treatment. A closely related form is the scalar projection of f onto some ϕ_m in the RKHS $\mathcal{H}$,

$$L_m[f] \triangleq \langle f, \phi_m \rangle_{\mathcal{H}}, \qquad (4)$$

which leads to $[\mathbf{k_u}(\mathbf{x})]_m = \phi_m(\mathbf{x})$ by the reproducing property of the RKHS. This, in effect, equips the GP approximation with basis functions ϕ_m that are not solely determined by the kernel, and suitable choices can lead to sparser representations and considerable computational benefits (Hensman et al., 2018; Burt et al., 2020; Dutordoir et al., 2020; Sun et al., 2021).

2.2. Spherical Harmonics Inducing Features

An instance of inter-domain features in the form of eq. 4 are the variational Fourier features (VFFs) (Hensman et al., 2018), in which ϕ_m form an orthogonal basis of trigonometric functions. This formulation offers significant computational advantages but scales poorly beyond a small handful

of dimensions. To address this, Dutordoir et al. (2020) propose a generalization of VFFs using the spherical harmonics for ϕ_m, which can be viewed as a multi-dimensional extension of the Fourier basis.

The construction relies on the Mercer decomposition of *zonal* kernels, which can be seen as the analog of stationary kernels in Euclidean spaces, but for hyperspheres. They can be expressed as $k(\mathbf{x}, \mathbf{x}') = \kappa(\tilde{\mathbf{x}}^\top \tilde{\mathbf{x}}')$ for some *shape function* $\kappa : [-1, 1] \to \mathbb{R}$, where $\tilde{\boldsymbol{\eta}} \triangleq \frac{\boldsymbol{\eta}}{\|\boldsymbol{\eta}\|} \in \mathbb{S}^{d-1}$ for any $\boldsymbol{\eta} \in \mathbb{R}^d$. Loosely speaking, just as stationary kernels are determined by the *distance* between inputs, zonal kernels depend only on the *angle* between inputs.

The spherical harmonics form an orthonormal basis on $L_2(\mathbb{S}^{d-1})$ consisting of the eigenfunctions of the kernel operator $\mathcal{K}$: $\mathcal{K}Y_{\ell,j} = a_\ell Y_{\ell,j}$, where $Y_{\ell,j}$ is the spherical harmonic of level ℓ and order j, and a_ℓ is the corresponding eigenvalue, or Fourier coefficient. Conveniently, by the Funk-Hecke theorem, a_ℓ can be computed by the one-dimensional integral

$$a_\ell = \frac{\Omega_d}{C_\ell^{(\alpha)}(1)} \int_{-1}^1 \kappa(t) C_\ell^{(\alpha)}(t)(1 - t^2)^{\frac{d-3}{2}} \, \mathrm{d}t,$$

where $C_\ell^{(\alpha)}$ is the Gegenbauer polynomial of degree ℓ and $\alpha \triangleq \frac{(d-1)}{2}$. Now, the number $J(d, \ell)$ of spherical harmonics that exist at a given level ℓ is determined by the multiplicity of eigenvalue a_ℓ. Thus, $\kappa(t)$ can be represented by

$$\kappa(t) = \|\boldsymbol{\xi}\|\|\boldsymbol{\xi}'\| \sum_{\ell=0}^\infty \sum_{j=1}^{J(d,\ell)} a_\ell Y_{\ell,j}(\tilde{\boldsymbol{\xi}}) Y_{\ell,j}(\tilde{\boldsymbol{\xi}}'), \quad (5)$$

where $t \triangleq \tilde{\boldsymbol{\xi}}^\top \tilde{\boldsymbol{\xi}}'$ for $\boldsymbol{\xi}, \boldsymbol{\xi}' \in \mathbb{R}^d$. Refer to the manuscript (Appendix B) of Dutordoir et al. (2021) for a concise summary of spherical harmonics in multiple dimensions.

Importantly, eq. 5 directly yields a Mercer decomposition for zonal kernels. In particular, let λ_ℓ denote the Fourier coefficients associated with kernel k. This gives rise to the inter-domain features $\phi_m \triangleq Y_{\ell,j}$, where m indexes the pairs (ℓ, j). Crucially, this leads to a diagonal covariance

$$[\mathbf{K_{uu}}]_{mm'} \triangleq \mathrm{Cov}(u_m, u_{m'}) = \lambda_m^{-1} \delta_{mm'},$$

where $\lambda_m \triangleq \lambda_\ell$ and δ denotes the Kronecker delta.

2.3. Neural Network Inducing Features

The recent works of Sun et al. (2020); Dutordoir et al. (2021) aim to construct inter-domain features ϕ_m such that $\mathbf{k_u}(\mathbf{x})$ in eq. 3 corresponds to a hidden layer in a feed-forward NN: $\sigma(\boldsymbol{\beta}\mathbf{x})$, for some $\boldsymbol{\beta} \in \mathbb{R}^{M \times d}$ and activation σ such as the SOFTPLUS or the rectified linear unit (RELU) function.
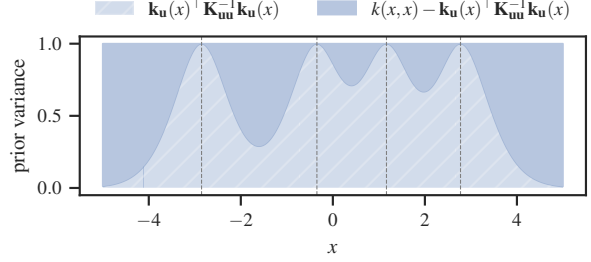

Figure 1. Prior variance deconstructed. The prior variance of $f(\mathbf{x})$ is $k(\mathbf{x}, \mathbf{x}) = \alpha$ for kernel amplitude $\alpha = 1$, which can be decomposed as the sum of the prior variances of $g(\mathbf{x})$ and $h(\mathbf{x})$. Vertical dashed lines indicate the location of inducing inputs $\mathbf{z}_m$ for $m = 1, \ldots, 4$. At these locations, the variance of $g(\mathbf{x})$ is 1 while that of $h(\mathbf{x})$ is 0.

In particular, let $H_m : \mathcal{X} \to \mathbb{R}$ denote the output of the m-th hidden unit. Additionally, let us project this function onto the unit hypersphere,

$$H_m(\mathbf{x}) \triangleq \|\mathbf{z}_m\|\|\mathbf{x}\| \cdot \sigma\left(\frac{\mathbf{z}_m^\top \mathbf{x}}{\|\mathbf{z}_m\|\|\mathbf{x}\|}\right). \quad (6)$$

Now, since this function is itself zonal, it can be represented in terms of the spherical harmonics as in eq. 5. Let ς_ℓ denote its associated Fourier coefficient. Thus, the inter-domain features can be defined as $\phi_m \triangleq H_m$, which leads to the covariance

$$[\mathbf{K_{uu}}]_{mm'} = \sum_{\substack{\ell=0: \\ \lambda_\ell \neq 0}}^\infty \frac{\varsigma_\ell^2}{\lambda_\ell} \frac{\ell + \alpha}{\alpha} C_\ell^{(\alpha)}\left(\frac{\mathbf{z}_m^\top \mathbf{z}_{m'}}{\|\mathbf{z}_m\|\|\mathbf{z}_{m'}\|}\right), \quad (7)$$

where λ_ℓ denotes the Fourier coefficients associated with kernel k. This formulation, which we refer to as ACTIVATED SVGP, has been shown to produce competitive results, especially when multiple ACTIVATED SVGP layers are stacked to form a DGP (Damianou & Lawrence, 2013), such that the propagation of the predictive means closely emulates a forward-pass through a deep NN.

Despite these favorable properties, ACTIVATED SVGPs have several limitations when it comes to their use with common covariance functions. Before elaborating on them in § 3, we discuss the orthogonally-decoupled GP framework on which our proposed extension relies.

2.4. Orthogonally Decoupled Inducing Points

Recent work has improved the efficiency of sparse GP methods through the structured decoupling of inducing variables (Cheng & Boots, 2017; Salimbeni et al., 2018; Shi et al., 2020). This not only enables the use of more variables at a reduced computational expense but also allows for more flexibility in modelling the predictive mean and covariance independently. We focus on the general framework of Shi et al. (2020) under which its predecessors can

Table 1. Summary of notation: relationships between input locations and output variables.

	W	**X**	**Z**
$f(\cdot)$	$\mathbf{v}$	$\mathbf{f}$	$\mathbf{u}$
$h(\cdot)$	$\mathbf{v}'$	$\mathbf{h}$	-

be subsumed. In particular, let the random function $f(\mathbf{x})$ from eq. 1 be decomposed into the sum of two independent GPs: $f(\mathbf{x}) = g(\mathbf{x}) + h(\mathbf{x})$, where

$$g(\mathbf{x}) \sim \mathcal{GP}(0, \mathbf{k}_{\mathbf{u}}^{\top}(\mathbf{x})\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{k}_{\mathbf{u}}(\mathbf{x}')), h(\mathbf{x}) \sim \mathcal{GP}(0, s(\mathbf{x}, \mathbf{x}'))$$

and let the covariance function $s(\mathbf{x}, \mathbf{x}')$ be defined according to the Schur complement of $\mathbf{K}_{\mathbf{uu}}$,

$$s(\mathbf{x}, \mathbf{x}') \triangleq k(\mathbf{x}, \mathbf{x}') - \mathbf{k}_{\mathbf{u}}^{\top}(\mathbf{x})\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{k}_{\mathbf{u}}(\mathbf{x}'),$$

where $\mathbf{k}_{\mathbf{u}}$ is defined in eq. 3. Intuitively, one can view g as the projection of f onto $\mathbf{u}$, and $h \perp g$, i.e. h is *orthogonal* to g (Hensman et al., 2018) in the statistical sense of linear independence (Rodgers et al., 1984). See Figure 1 for an illustration of the priors of $g(\mathbf{x})$ and $h(\mathbf{x})$. Let $\mathbf{h}$ be the values of h at observed inputs $\mathbf{X}$, i.e. $\mathbf{h} \triangleq h(\mathbf{X})$. Then we have $p(\mathbf{h}) = \mathcal{N}(\mathbf{0}, \mathbf{S}_{\mathbf{ff}})$, where $\mathbf{S}_{\mathbf{ff}} \triangleq \mathbf{K}_{\mathbf{ff}} - \mathbf{Q}_{\mathbf{ff}}$. This allows one to reparameterize $\mathbf{f} \sim p(\mathbf{f} \mid \mathbf{u})$ from eq. 2, for a given $\mathbf{u}$, as

$$\mathbf{f} = \mathbf{Q}_{\mathbf{fu}}\mathbf{u} + \mathbf{h}, \qquad \mathbf{h} \sim p(\mathbf{h}). \tag{8}$$

The model's joint distribution can now be written as $p(\mathbf{y}, \mathbf{h}, \mathbf{u}) = p(\mathbf{y} \mid \mathbf{h}, \mathbf{u})p(\mathbf{h})p(\mathbf{u})$, where $p(\mathbf{y} \mid \mathbf{h}, \mathbf{u}) = \mathcal{N}(\mathbf{y} \mid \mathbf{Q}_{\mathbf{fu}}\mathbf{u} + \mathbf{h}, \beta^{-1}\mathbf{I})$. Next, *orthogonal* inducing variables $\mathbf{v}$, which represent the values of f at a collection of K orthogonal inducing locations $\mathbf{W} \triangleq [\mathbf{w}_1 \cdots \mathbf{w}_K]^{\top}$, are introduced. Similarly, inducing variables $\mathbf{v}'$ represent the values of h at $\mathbf{W}$. The reader may find it helpful to refer to Table 1 for a summary of the relationships between the input locations and the output variables defined thus far.

Now, by definition, $\mathbf{v} = \mathbf{Q}_{\mathbf{vu}}\mathbf{u} + \mathbf{v}'$, where $\mathbf{Q}_{\mathbf{vu}} \triangleq \mathbf{K}_{\mathbf{vu}}\mathbf{K}_{\mathbf{uu}}^{-1}$ which is analogous to the relationship between $\mathbf{f}$ and $\mathbf{h}$ in eq. 8. Therefore, one need only be concerned with the treatment of $\mathbf{v}'$. The joint distribution of the model augmented by the variables $\mathbf{v}'$ now becomes $p(\mathbf{y}, \mathbf{h}, \mathbf{u}, \mathbf{v}') = p(\mathbf{y} \mid \mathbf{h}, \mathbf{u})p(\mathbf{u})p(\mathbf{h}, \mathbf{v}')$, where $p(\mathbf{h}, \mathbf{v}') = p(\mathbf{h} \mid \mathbf{v}')p(\mathbf{v}')$ for $p(\mathbf{v}') = \mathcal{N}(\mathbf{0}, \mathbf{S}_{\mathbf{vv}})$ and $p(\mathbf{h} \mid \mathbf{v}') = \mathcal{N}(\mathbf{h} \mid \mathbf{S}_{\mathbf{fv}}\mathbf{S}_{\mathbf{vv}}^{-1}\mathbf{v}', \mathbf{S}_{\mathbf{ff}} - \mathbf{S}_{\mathbf{fv}}\mathbf{S}_{\mathbf{vv}}^{-1}\mathbf{S}_{\mathbf{vf}})$, with

$$\mathbf{S}_{\mathbf{vf}} \triangleq \mathbf{K}_{\mathbf{vf}} - \mathbf{Q}_{\mathbf{vf}}, \quad \mathbf{Q}_{\mathbf{vf}} \triangleq \mathbf{Q}_{\mathbf{vu}}\mathbf{K}_{\mathbf{uu}}\mathbf{Q}_{\mathbf{uf}}, \tag{9}$$

$$\mathbf{S}_{\mathbf{vv}} \triangleq \mathbf{K}_{\mathbf{vv}} - \mathbf{Q}_{\mathbf{vv}}, \quad \mathbf{Q}_{\mathbf{vv}} \triangleq \mathbf{Q}_{\mathbf{vu}}\mathbf{K}_{\mathbf{uu}}\mathbf{Q}_{\mathbf{uv}}. \tag{10}$$

Let the variational distribution now be $q(\mathbf{h}, \mathbf{u}, \mathbf{v}') = p(\mathbf{h} \mid \mathbf{v}')q(\mathbf{u}, \mathbf{v}')$, where $q(\mathbf{u}, \mathbf{v}') \triangleq q(\mathbf{u})q(\mathbf{v}')$ and $q(\mathbf{v}') \triangleq \mathcal{N}(\mathbf{m}_{\mathbf{v}}, \mathbf{C}_{\mathbf{v}})$ for variational parameters $\mathbf{m}_{\mathbf{v}} \in \mathbb{R}^{K}$ and

$\mathbf{C}_{\mathbf{v}} \in \mathbb{R}^{K \times K}$ s.t. $\mathbf{C}_{\mathbf{v}} \succeq 0$. This gives the posterior predictive density $q(\mathbf{f}_*) = \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_{**})$, where

$$\boldsymbol{\mu}_* \triangleq \mathbf{Q}_{*\mathbf{u}}\mathbf{m}_{\mathbf{u}} + \mathbf{S}_{*\mathbf{v}}\mathbf{S}_{\mathbf{vv}}^{-1}\mathbf{m}_{\mathbf{v}}, \tag{11}$$

$$\begin{aligned}\boldsymbol{\Sigma}_{**} \triangleq{} & \mathbf{K}_{**} + \mathbf{Q}_{*\mathbf{u}}(\mathbf{C}_{\mathbf{u}} - \mathbf{K}_{\mathbf{uu}})\mathbf{Q}_{\mathbf{u}*} \\ & + \mathbf{S}_{*\mathbf{v}}\mathbf{S}_{\mathbf{vv}}^{-1}(\mathbf{C}_{\mathbf{v}} - \mathbf{S}_{\mathbf{vv}})\mathbf{S}_{\mathbf{vv}}^{-1}\mathbf{S}_{\mathbf{v}*}.\end{aligned} \tag{12}$$

Thus seen, prediction incurs a cost of $\mathcal{O}(M^3 + K^3)$ in this framework. Like the so-called ODVGP framework of Salimbeni et al. (2018), when seen from the dual RKHS perspective, the predictive mean can be decomposed into a component that shares the same standard basis as the covariance, in addition to another component that is *orthogonal* to the standard basis. However, this framework extends ODVGP further by also decomposing the predictive covariance into parts corresponding to the standard and orthogonal bases. Accordingly, setting $\mathbf{C}_{\mathbf{v}} = \mathbf{S}_{\mathbf{vv}}$ recovers the ODVGP framework, and further setting $\mathbf{m}_{\mathbf{v}} = \mathbf{0}$ recovers the standard SVGP framework.

3. Methodology

We begin this section by outlining some of the limitations of ACTIVATED SVGPs that preclude the use of numerous kernels and inducing features, not the least of which being popular choices of kernels such as the squared exponential (SE) kernel and the Matérn family of kernels, combined with NN inducing features with RELU activations.

The root cause of these issues can be seen in Figure 2, where the Fourier coefficients of various combinations of kernels and activation features are visualized. Specifically, for each combination, we compare the (root of the) kernel coefficients $\sqrt{\lambda_\ell}$ against the feature coefficients ς_ℓ at increasing levels $\ell = 1, \ldots, 35$. The posterior predictives that result from fitting ACTIVATED SVGP models with these combinations are shown in Figure 3. We consider the Matérn-$5/2$ kernel as our running example, but the analysis extends to all stationary kernels.

Spectra mismatch. For the Matérn kernel (left column of panes in Figures 2 and 3), we see that there are multiple levels ℓ at which the feature coefficients are zero while the corresponding kernel coefficients are nonzero. Such discrepancies in the spectra yields a poor Nyström approximation $\mathbf{Q}_{\mathbf{ff}}$ that fails to fully capture the prior covariance $\mathbf{K}_{\mathbf{ff}}$ induced by the kernel, which subsequently leads to the overestimation of the predictive variance and therefore a suboptimal evidence lower bound (ELBO). In contrast, the Arccos kernel does not suffer from this pathology.

RKHS inner product. The RKHS inner product associated with zonal kernels in general is a series consisting of ratios of Fourier coefficients. Since the RELU feature coefficients (top row of panes in Figures 2 and 3) decay at the same rate as the square root of the kernel coefficients, this results
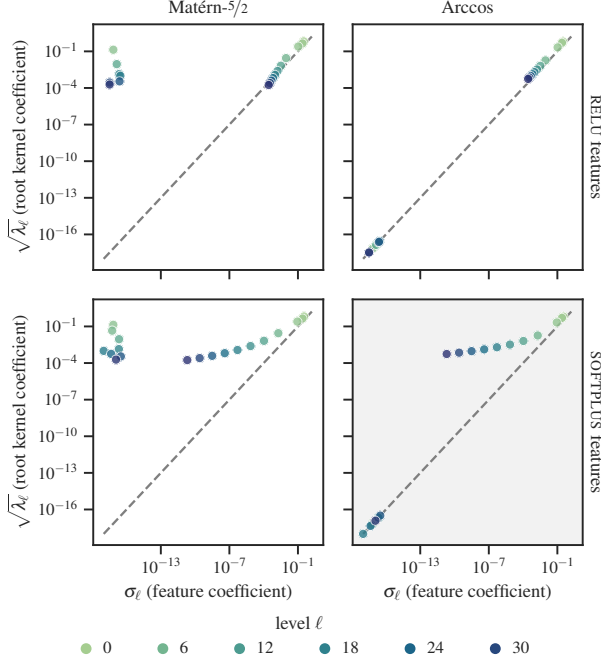
Figure 2. Comparison of the Fourier coefficients of various kernels and activation features for increasing levels $\ell = 1, \ldots, 35$.

in a divergent series which in turn renders the RKHS inner product indeterminate. In contrast, the feature coefficients of the comparatively smoother SOFTPLUS activation (bottom row of panes in Figures 2 and 3) decay at a much faster rate, and thus yields a well-defined RKHS inner product. For the reasons outlined above, the work of Dutordoir et al. (2021) restricted its scope to the use of the Arccos kernel in conjunction with the SOFTPLUS activation (pane highlighted in gray in Figure 2).

Truncation error. Lastly, as expected, the truncation of the series in eq. 7 at some finite number L of spherical harmonic levels often leads to overly smooth predictive response surfaces and overestimation of the variance.

Spherical Features for Orthogonally-Decoupled GPs

We propose extending the orthogonally-decoupled GP framework (§ 2.4) to use inter-domain inducing features. Accordingly, let $u_m \triangleq \langle f, \phi_m \rangle_{\mathcal{H}}$ and $v_k \triangleq \langle f, \psi_k \rangle_{\mathcal{H}}$ for some arbitrary choices of $\phi_m, \psi_k \in \mathcal{H}$. This generalizes the framework of Shi et al. (2020) since, by the reproducing property, setting $\phi_m : \mathbf{x} \mapsto k(\mathbf{z}_m, \mathbf{x})$ and $\psi_k : \mathbf{x} \mapsto k(\mathbf{w}_k, \mathbf{x})$ leads to standard inducing points, $u_m = f(\mathbf{z}_m), v_k = f(\mathbf{w}_k)$.

In particular, we define $\phi_m \triangleq H_m$, the m-th unit of the spherical activation layer (eq. 6) described in § 2.3, and $\psi_k(\mathbf{x}) \triangleq k(\mathbf{w}_k, \mathbf{x})$. The posterior predictive of the model described in § 2.4, summarized by Equations (11) and (12), is fully determined by the covari-
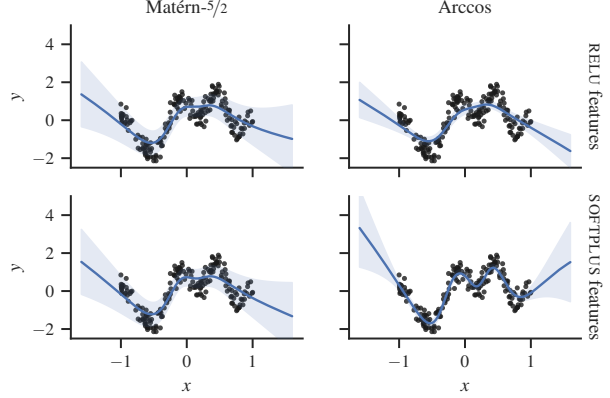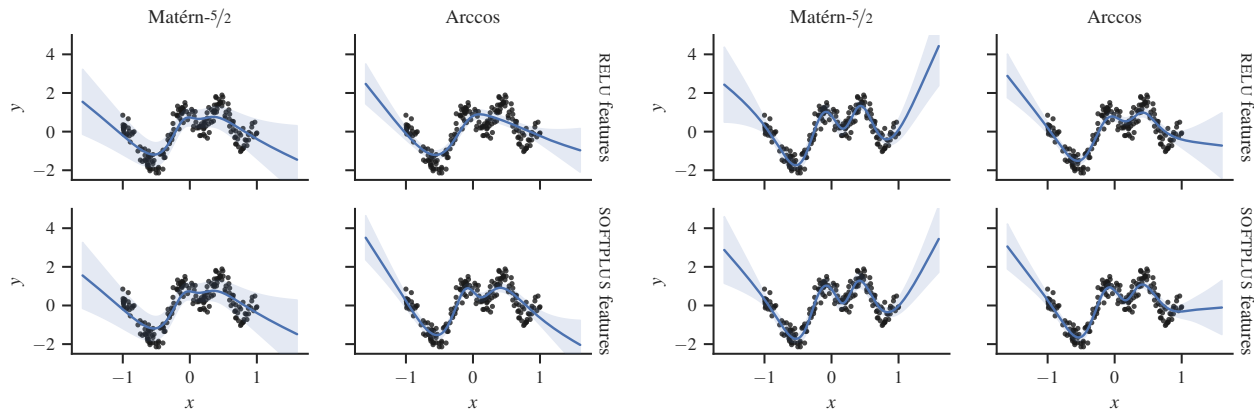

Figure 3. Posterior predictives of ACTIVATED SVGP models various kernels and activation features and $L = 16$ levels.

ances $\mathbf{K_{ff}}, \mathbf{K_{uf}}, \mathbf{K_{vf}}, \mathbf{K_{uu}}, \mathbf{K_{vu}}$ and $\mathbf{K_{vv}}$. Recall that $[\mathbf{K_{uf}}]_{mn} = [\mathbf{k_u}(\mathbf{x}_n)]_m$ and $\mathbf{K_{uu}}$ is precisely as expressed in eq. 7. We have

$$[\mathbf{K_{vf}}]_{kn} \triangleq \mathrm{Cov}\,(v_k, f(\mathbf{x}_n)) = k(\mathbf{w}_k, \mathbf{x}_n),$$
$$[\mathbf{K_{vu}}]_{km} \triangleq \mathrm{Cov}\,(v_k, u_m) = \phi_m(\mathbf{w}_k),$$
$$[\mathbf{K_{vv}}]_{kk'} \triangleq \mathrm{Cov}\,(v_k, v_{k'}) = k(\mathbf{w}_k, \mathbf{w}_{k'}).$$

Note that the cross-covariance $\mathbf{K_{vu}}$ between $\mathbf{u}$ and $\mathbf{v}$ can be interpreted as the forward-pass of the orthogonal pseudo-input $\mathbf{w}_k$ through the NN activation H_m. Crucially, these terms constitute the orthogonal basis and provide additional degrees of flexibility, through free parameters $\mathbf{W}$, that can compensate for errors remaining from the original basis—in both the predictive mean and variance. Suffice it to say, this is not the only possible choice but is one that possesses a number of appealing properties. We compare against a few other possibilities which we enumerate in Appendix A.

As discussed in § 2.4, the addition of K inducing variables incurs a cost of $\mathcal{O}(M^3 + K^3)$. More precisely: suppose the exact cost is $C \cdot (M^3 + K^3)$ operations for some constant C w.r.t. M, K. Further, suppose $K = B \cdot M$ for some $B > 0$. Then there are a total of $(B + 1) \cdot M$ inducing variables (orthogonal or otherwise) and the cost becomes $(B^3 + 1)C \cdot M^3$. By comparison, incorporating the same number of inducing variables in SVGP costs $(B+1)^3 C \cdot M^3$. That is, this approach leads to a (B^3+1)-fold increase in the constant rather than a $(B + 1)^3$-fold increase. Concretely, this means that doubling the number of inducing variables doubles the constant in this approach, but leads to an *eight-fold* increase in SVGP. While such a difference vanishes asymptotically for large M and K, it still has a considerable impact for modest sizes ($M, K < 1,000$) that are feasible in practice. Thus seen, incorporating an orthogonal basis spanned by K inducing variables is a more cost-effective strategy for improving ACTIVATED SVGP than increasing M or the truncation level L.

(a) Inducing activation features with $L = 8$ levels.

(b) Inducing activation features with $L = 8$ levels and $K = 8$ orthogonal bases (*our method*).

Figure 4. Posterior predictives of ACTIVATED SVGP with various kernels and activation features on the 1D Snelson dataset; *black circular markers* represent the observations; *blue solid lines* and *shaded regions* denote the mean and the ± 2 standard deviations, resp.

4. Experiments

We describe the experiments conducted to empirically validate our approach. The open-source implementation of our method can be found at: ltiao/spherical-orthogonal-gaussian-processes. Further information concerning the experimental set-up and various implementation details can be found in Appendix B.

4.1. Synthetic 1D Dataset

We highlight some notable properties of our method on the one-dimensional dataset of Snelson & Ghahramani (2007).

First we fit ACTIVATED SVGP models with different combinations of kernels and activation features using $L = 8$ truncation levels. The resulting posterior predictives are shown in Figure 4. More specifically, in Figure 4a, we see that none of the model fits are particularly tight due in part to truncation errors, since we are using relatively few levels. This is especially true of the Matérn kernel (left column of panes), which results in a posterior that is not only too smooth but also clearly suffering from an overestimation of the variance. A conceptually straightforward way to improve performance is to increase the truncation level. Accordingly, Figure 3 (introduced earlier in § 3) showed results from effectively the exact same set-up, but with twice the number of levels ($L = 16$). With this increase, we see a clear improvement in the Arccos-SOFTPLUS case, but no discernible difference in the other combinations. Notably, the overestimation of the variances in the Matérn kernel persists. By comparison, Figure 4b shows results from using $L = 8$ truncation levels, but with the addition of $K = 8$ orthogonal inducing variables. Remarkably, incorporating just a handful of these variables produces substantial improvements,

not least for the Matérn kernel.

Figure 5 offers a deeper insight into the underlying mechanisms that contribute to these improvements. Here we plot the predictive variance (eq. 12) in terms of its constituent parts. In Figure 5a, we see that the variance estimate with Matérn kernels is heavily distorted by large spurious contributions in the $\mathbf{K}_{\mathbf{ff}} - \mathbf{Q}_{\mathbf{ff}}$ term (*dark blue solid line*), which is caused by the pathology described in § 3. On the other hand, in Figure 5b, such spurious contributions also appear, but are offset by the subtractive term $\mathbf{S}_{\mathbf{fv}}\mathbf{S}_{\mathbf{vv}}^{-1}\mathbf{S}_{\mathbf{vf}}$ (*dark orange dashed line*). This term constitutes the orthogonal basis, and provides added flexibility that is effective at nullifying errors introduced by the original basis.

Each of the three variations discussed above are repeated 5 times, and some quantitative results are summarized in Figure 6. Specifically, we report the ELBO and the *throughput*, i.e. the average number of optimization iterations completed per second. The ACTIVATED SVGP with $L = 8$ truncation levels, as seen in Figures 4a and 5a, is represented by the *blue circular markers*. The model resulting from doubling the number of levels $L = 16$, as seen in Figure 3, is represented by the *orange circular markers*. As discussed, this leads to an improvement in the Arccos-SOFTPLUS case, but to modest or no improvements otherwise. However, we can now see that this has come at a significant computational expense, as the throughput has reduced by roughly half. On the other hand, the model resulting from retaining the same truncation level but incorporate an orthogonal basis consisting of $K = 8$ variables, as seen in Figures 4b and 5b, is represnted by the *blue cross markers*. This can be seen to have roughly the same footprint as doubling the truncation level, but leads to a considerably improved model fit, especially in cases involving the Matérn kernel (the only
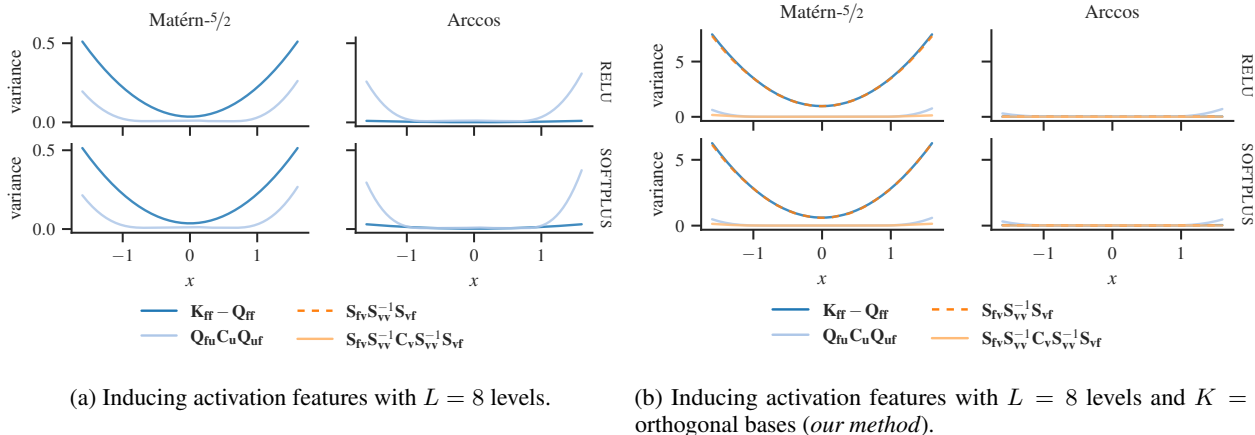
(a) Inducing activation features with $L = 8$ levels.

(b) Inducing activation features with $L = 8$ levels and $K = 8$ orthogonal bases (*our method*).

Figure 5. Decomposition of the posterior predictive variances of SVGP with various kernels and activation features on the 1D SNELSON dataset (see Figure 4) into its constituent terms; the additive terms that constitute the predictive variance are indicated by *solid* lines, while the subtractive terms are indicated by *dashed* lines; terms that constitute the predictive variance of the original SVGP model (Titsias, 2009) have a *blue* hue, while additional terms introduced by the orthogonally-decoupled model (Shi et al., 2020) have an *orange* hue.
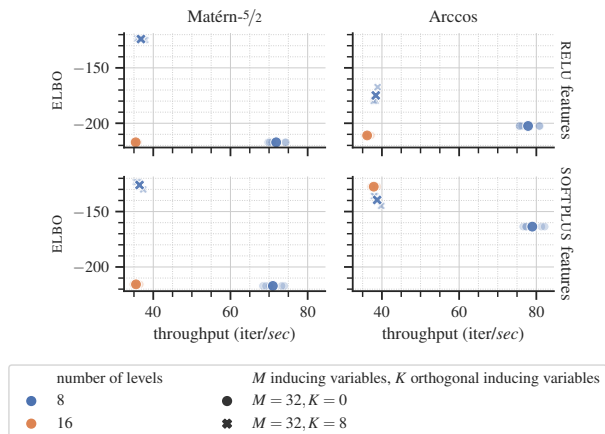
Figure 6. The ELBO and throughput (of model fitting) for various kernels and activation features and the configurations visualized in Figure 3, Figures 4a and 5a, and Figures 4b and 5b; markers with low opacity represent the individual runs, while markers with high opacity represent the mean of each group.

exception is in the Arccos-SOFTPLUS case, where doubling the truncation level retains a slight advantage). All told, incorporating an orthogonal basis has roughly the same cost as doubling the truncation level but leads to significantly better performance improvements.

4.2. Regression on UCI Repository Datasets

We evaluate our method on a number of well-studied regression problems from the UCI repository of datasets (Dua & Graff, 2017). In particular, we consider the YACHT, CONCRETE, ENERGY, KIN8NM and POWER datasets. Additional results on the larger datasets from this collection can be found in Appendix C.2.

We fit variations of SVGP with the Arccos, Matérn, and SE kernels, and (a) standard inducing points, and inter-domain inducing features based on (b) RELU- and (c) SOFTPLUS-activated inducing features. For each of these variants, we consider three combinations of base and orthogonal inducing variables: (i-ii) 128 and 256 base inducing variables (and no orthogonal inducing variables), and (iii) 128 base inducing variables with 128 orthogonal inducing variables. The activation features are truncated at $L = 6$ levels. Our proposed method is represented by the combinations consisting of RELU- and SOFTPLUS-activated features with orthogonal inducing variables (b-c,iii). The remaining combinations, against which we benchmark, correspond to the original SVGP (a,i-ii) (Titsias, 2009), SOLVEGP (a,iii) (Shi et al., 2020), and ACTIVATED SVGP (b-c,i-ii) (Dutordoir et al., 2021).

To quantitatively assess performance, we report the test root-mean-square error (RMSE) and negative log predictive density (NLPD), shown in Figures 7 and 9, respectively. Unless otherwise stated, for each method and problem, we perform random sub-sampling validation by aggregating results from 5 repetitions across 10% held-out test sets. Within the training set, the inputs and outputs are standardized, i.e. scaled to have zero mean and unit variance and subsequently restored to the original scale at test time.

We observe that, irrespective of the choice of kernel, when using activation features, whether RELU- or SOFTPLUS-activated, augmenting the model with orthogonal bases significantly improves performance, notably even more so than doubling the number of base inducing variables. This can readily be seen across all datasets on both the NLPD and RMSE metrics. Further, with the Arccos kernel, it outperforms its counterparts based on standard inducing
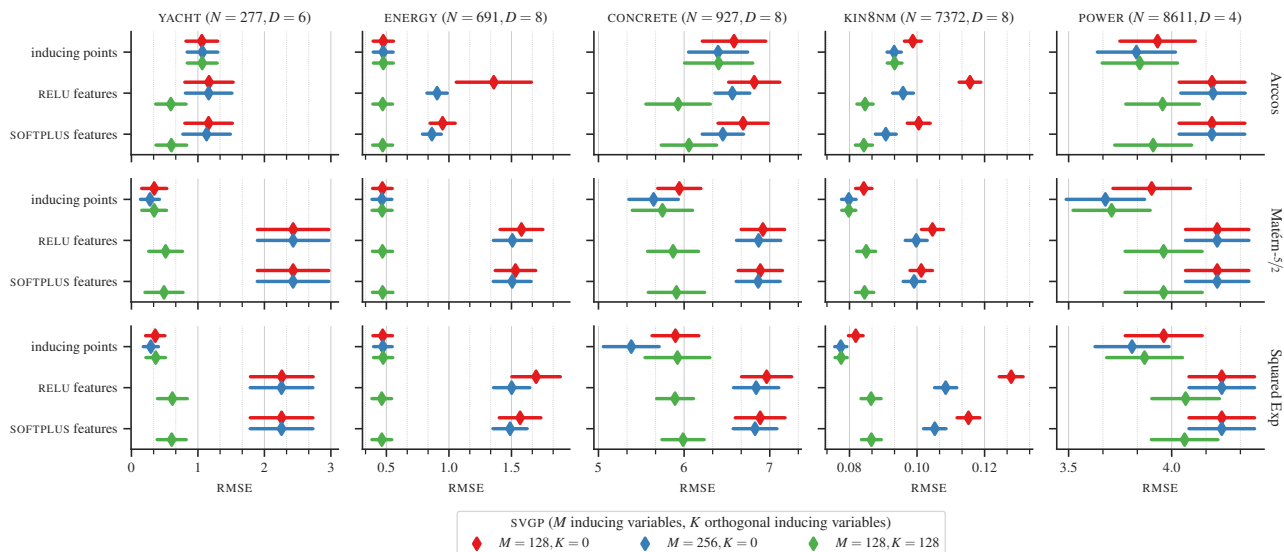
Figure 7. Test RMSE on regression problems from the UCI repository of datasets for various kernels and activation features. Along the rows labeled *"inducing points"*, the red and blue markers (♦, ♦) represent the original SVGP model (Titsias, 2009), while the green markers (♦) represent SOLVEGP (Shi et al., 2020). Along the remaining rows, the red and blue markers (♦, ♦) represent the ACTIVATED SVGP (Dutordoir et al., 2021), while the green markers (♦) represent our proposed approach.
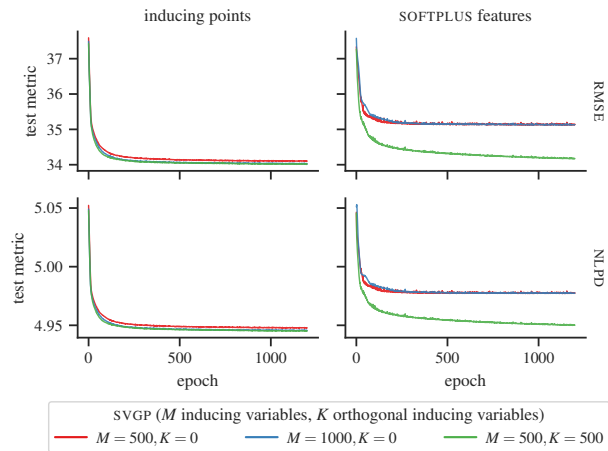
Figure 8. Test metrics, RMSE and NLPD, on the large-scale 2008 U.S. airline delays dataset using the *Arccos* kernel with standard inducing points and SOFTPLUS-activated features. Along the column labeled "inducing points", the red and blue lines (— and —) represent the mini-batch SVGP (Hensman et al., 2013), while the green line (—) represents SOLVEGP (Shi et al., 2020). Along the column labeled "SOFTPLUS features", the red and blue lines (— and —) represent the ACTIVATED SVGP (Dutordoir et al., 2021), while the green line (—) represents our proposed approach.

points across most datasets (the exception being the POWER dataset). With the Matérn and SE kernels, it achieves results comparable to its standard inducing points counterparts in most datasets.

4.3. Large-scale Regression on Airline Delays Dataset

Finally, we consider a large-scale regression dataset concerning U.S. commercial airline delays in 2008. The task is to forecast the duration of delays in reaching the destination of a given flight, utilizing information such as the route distance, airtime, scheduled month, day of the week, and other relevant factors, as well as characteristics of the aircraft such as its age (number of years since deployment). The complete dataset encompasses 5,929,413 flights, of which we randomly select 1M observations without replacement to form a subset that is more manageable but still considerable in scale. Results on a reduced 100K subset can be found in Appendix C.1.

To quantitatively assess performance, we report the test RMSE and NLPD evaluated on a $^1/_3$ held-out test set. The results are shown in the top and bottom rows of Figure 8, respectively. Within the training set, the inputs and outputs are standardized, i.e. scaled to have zero mean and unit variance and subsequently restored to the original scale at test time.

Given the immense volume of data at hand, we are compelled to utilize mini-batch training for stochastic optimisation (Hensman et al., 2013). To this end, we use the Adam optimizer (Kingma & Ba, 2014) with its typical default settings (learning rate 1×10^{-3}, $\beta_1 = 0.9$, $\beta_2 = 0.999$). Our batch size is set to 5,000, and we train the models for a total of 1,200 epochs.
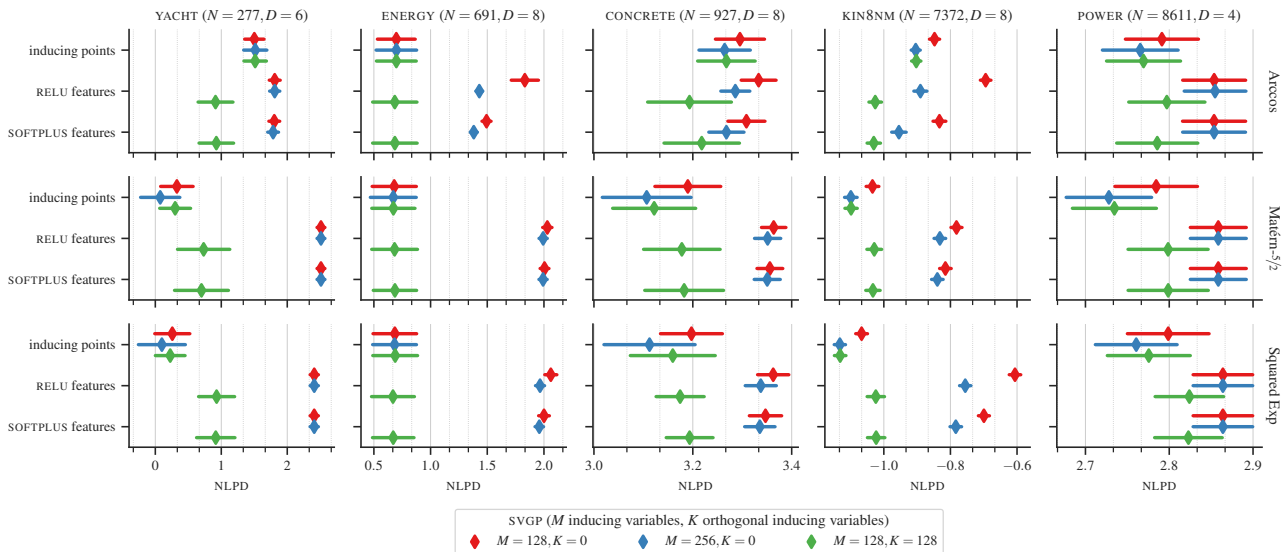
Figure 9. Test NLPD on regression problems from the UCI repository of datasets for various kernels and activation features. Along the rows labeled *"inducing points"*, the red and blue markers (◆, ◆) represent the original SVGP model (Titsias, 2009), while the green markers (◆) represent SOLVEGP (Shi et al., 2020). Along the remaining rows, the red and blue markers (◆, ◆) represent the ACTIVATED SVGP (Dutordoir et al., 2021), while the green markers (◆) represent our proposed approach.

We fit variations of SVGP with the Arccos kernel and (a) standard inducing points and (b) inter-domain inducing features based on SOFTPLUS-activated inducing features. For each of these variants, we consider three combinations of base and orthogonal inducing variables: (i-ii) 500 and 1,000 base inducing variables (and no orthogonal inducing variables), and (iii) 500 base inducing variables with 500 orthogonal inducing variables. The activation features are truncated at $L = 6$ levels. Our proposed method is represented by the combination consisting of SOFTPLUS-activated features with orthogonal inducing variables (b,iii). The remaining combinations, against which we benchmark, correspond to the mini-batch SVGP (a,i-ii) (Hensman et al., 2013), SOLVEGP (a,iii) (Shi et al., 2020), and ACTIVATED SVGP (b,i-ii) (Dutordoir et al., 2021).

The outcomes are as expected when employing standard inducing points (left). In particular, doubling the number of base inducing points from 500 to 1,000 demonstrates significant improvements. Furthermore, by using 500 base inducing points alongside 500 orthogonal inducing points, we achieve comparable performance to having 1,000 base inducing points, while enjoying improved computationally efficiency. In contrast, when examining the ACTIVATED SVGP model with SOFTPLUS features (right), it's apparent that it underperforms compared to the original SVGP counterparts. Furthermore, doubling the number of inducing features from 500 to 1,000 has virtually no effect. However, by incorporating orthogonal bases into the ACTIVATED

SVGP model with 500 features following our proposed approach, we witness substantial improvements and achieve comparable performance to its standard inducing points counterparts.

5. Conclusion

We considered the use of inter-domain inducing features in the orthogonally-decoupled SVGP framework, specifically, the spherical activation features, and showed that this alleviates some of the practical issues and shortcomings associated with the ACTIVATED SVGP model. We demonstrated the effectiveness of this approach by conducting empirical evaluations on several problems, and showed that this leads to enhanced predictive performance over more computationally demanding alternatives such as increasing the truncation levels or the number of inducing variables.

Future work will explore alternative designs of inter-domain inducing features to construct new standard and orthogonal bases that provide additional complementary benefits.

Acknowledgements

References

Burt, D. R., Rasmussen, C. E., and van der Wilk, M. Variational orthogonal features. *arXiv preprint arXiv:2006.13170*, 2020.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing*, 16(5):1190–1208, 1995.

Calandra, R., Peters, J., Rasmussen, C. E., and Deisenroth, M. P. Manifold Gaussian processes for regression. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 3338–3345. IEEE, 2016.

Cheng, C.-A. and Boots, B. Variational inference for Gaussian process models with linear complexity. *Advances in Neural Information Processing Systems*, 30, 2017.

Csató, L. and Opper, M. Sparse on-line Gaussian processes. *Neural computation*, 14(3):641–668, 2002.

Damianou, A. and Lawrence, N. D. Deep Gaussian processes. In *Artificial intelligence and statistics*, pp. 207–215. PMLR, 2013.

Deisenroth, M. and Rasmussen, C. E. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pp. 465–472, 2011.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Dutordoir, V., Durrande, N., and Hensman, J. Sparse Gaussian processes with spherical harmonic features. In *International Conference on Machine Learning*, pp. 2793–2802. PMLR, 2020.

Dutordoir, V., Hensman, J., van der Wilk, M., Ek, C. H., Ghahramani, Z., and Durrande, N. Deep neural networks as point estimates for deep Gaussian processes. *Advances in Neural Information Processing Systems*, 34, 2021.

Garnett, R. *Bayesian Optimization*. Cambridge University Press, 2023. to appear.

Hennig, P., Osborne, M. A., and Kersting, H. P. *Probabilistic Numerics*. Cambridge University Press, 2022.

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*, 2013.

Hensman, J., Matthews, A., and Ghahramani, Z. Scalable variational Gaussian process classification. In *Artificial Intelligence and Statistics*, pp. 351–360. PMLR, 2015a.

Hensman, J., Matthews, A. G., Filippone, M., and Ghahramani, Z. MCMC for variationally sparse Gaussian processes. *Advances in Neural Information Processing Systems*, 28, 2015b.

Hensman, J., Durrande, N., and Solin, A. Variational Fourier features for Gaussian processes. *Journal of Machine Learning Research*, 18(151):1–52, 2018. URL http://jmlr.org/papers/v18/16-579.html.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lázaro-Gredilla, M. and Figueiras-Vidal, A. Inter-domain Gaussian processes for sparse inference using inducing features. *Advances in Neural Information Processing Systems*, 22, 2009.

MacKay, D. J. et al. Introduction to Gaussian processes. *NATO ASI series F computer and systems sciences*, 168: 133–166, 1998.

Matthews, A. G. d. G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. GPflow: A Gaussian process library using TensorFlow. *Journal of Machine Learning Research*, 18(40):1–6, apr 2017. URL http://jmlr.org/papers/v18/16-537.html.

Murray, I. and Adams, R. P. Slice sampling covariance hyperparameters of latent Gaussian models. *Advances in neural information processing systems*, 23, 2010.

Neal, R. M. Bayesian learning for neural networks, 1996.

Ober, S. W., Rasmussen, C. E., and van der Wilk, M. The promises and pitfalls of deep kernel learning. In *Uncertainty in Artificial Intelligence*, pp. 1206–1216. PMLR, 2021.

Quinonero-Candela, J. and Rasmussen, C. E. A unifying view of sparse approximate Gaussian process regression. *The Journal of Machine Learning Research*, 6: 1939–1959, 2005.

Rasmussen, C. E. and Williams, C. K. *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006.

Rodgers, J. L., Nicewander, W. A., and Toothaker, L. Linearly independent, orthogonal, and uncorrelated variables. *The American Statistician*, 38(2):133–134, 1984.

Salimbeni, H., Cheng, C.-A., Boots, B., and Deisenroth, M. Orthogonally decoupled variational Gaussian processes. *Advances in Neural Information Processing Systems*, 31, 2018.

Seeger, M. W., Williams, C. K., and Lawrence, N. D. Fast forward selection to speed up sparse Gaussian process regression. In *International Workshop on Artificial Intelligence and Statistics*, pp. 254–261. PMLR, 2003.

Shi, J., Titsias, M., and Mnih, A. Sparse orthogonal variational inference for Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, pp. 1932–1942. PMLR, 2020.

Snelson, E. and Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. *Advances in neural information processing systems*, 18, 2005.

Snelson, E. and Ghahramani, Z. Local and global sparse Gaussian process approximations. In *Artificial Intelligence and Statistics*, pp. 524–531. PMLR, 2007.

Sun, S., Shi, J., and Grosse, R. B. Neural networks as inter-domain inducing points. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2020.

Sun, S., Shi, J., Wilson, A. G. G., and Grosse, R. B. Scalable variational Gaussian processes via harmonic kernel decomposition. In *International Conference on Machine Learning*, pp. 9955–9965. PMLR, 2021.

Titsias, M. Variational learning of inducing variables in sparse Gaussian processes. In *Artificial Intelligence and Statistics*, pp. 567–574. PMLR, 2009.

van der Wilk, M., Dutordoir, V., John, S., Artemev, A., Adam, V., and Hensman, J. A framework for interdomain and multioutput Gaussian processes. *arXiv preprint arXiv:2003.01115*, 2020.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. Deep kernel learning. In *Artificial intelligence and statistics*, pp. 370–378. PMLR, 2016.

Zhu, C., Byrd, R. H., Lu, P., and Nocedal, J. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on mathematical software (TOMS)*, 23(4):550–560, 1997.

A. Combinations

Recall that the prior over $\mathbf{f}, \mathbf{u}, \mathbf{v}$ is

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{u} \\ \mathbf{v} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K_{ff}} & \mathbf{K_{uf}^\top} & \mathbf{K_{vf}^\top} \\ \mathbf{K_{uf}} & \mathbf{K_{uu}} & \mathbf{K_{vu}^\top} \\ \mathbf{K_{vf}} & \mathbf{K_{vu}} & \mathbf{K_{vv}} \end{bmatrix} \right) \tag{13}$$

Other than the variational parameters, the ingredients necessary to compute the predictive for the model described in § 2.4, summarized by Equations (11) and (12), are

$$\mathbf{S_{ff}} \triangleq \mathbf{K_{ff}} - \mathbf{Q_{ff}}, \qquad \mathbf{S_{vf}} \triangleq \mathbf{K_{vf}} - \mathbf{Q_{vf}}, \qquad \mathbf{S_{vv}} \triangleq \mathbf{K_{vv}} - \mathbf{Q_{vv}},$$

where

$$\mathbf{Q_{ff}} \triangleq \mathbf{Q_{fu}K_{uu}Q_{uf}}, \qquad \mathbf{Q_{vf}} \triangleq \mathbf{Q_{vu}K_{uu}Q_{uf}}, \qquad \mathbf{Q_{vv}} \triangleq \mathbf{Q_{vu}K_{uu}Q_{uv}},$$

and

$$\mathbf{Q_{fu}} \triangleq \mathbf{K_{fu}K_{uu}^{-1}}, \qquad \mathbf{Q_{vu}} \triangleq \mathbf{K_{vu}K_{uu}^{-1}}.$$

In all the cases we shall discuss below, $\mathbf{K_{ff}}$ will evaluate to

$$[\mathbf{K_{ff}}]_{nn'} = \mathrm{Cov}\left(f(\mathbf{x}_n), f(\mathbf{x}_{n'})\right) = \langle k(\mathbf{x}_n, \cdot), k(\mathbf{x}_{n'}, \cdot) \rangle_{\mathcal{H}} = k(\mathbf{x}_n, \mathbf{x}_{n'}). \tag{14}$$

Similarly, $\mathbf{K_{uf}}$ and $\mathbf{K_{vf}}$ will always simplify to

$$[\mathbf{K_{uf}}]_{mn} = \mathrm{Cov}\left(u_m, f(\mathbf{x}_n)\right) = \langle \phi_m, k(\mathbf{x}_n, \cdot) \rangle_{\mathcal{H}} = \phi_m(\mathbf{x}_n) \tag{15}$$
$$[\mathbf{K_{vf}}]_{kn} = \mathrm{Cov}\left(v_k, f(\mathbf{x}_n)\right) = \langle \psi_k, k(\mathbf{x}_n, \cdot) \rangle_{\mathcal{H}} = \psi_k(\mathbf{x}_n) \tag{16}$$

The remaining covariances $\mathbf{K_{uu}}, \mathbf{K_{vu}}$ and $\mathbf{K_{vv}}$ reduce to different forms, depending on the choices of ϕ_m and ψ_k,

$$[\mathbf{K_{uu}}]_{mm'} = \mathrm{Cov}\left(u_m, u_{m'}\right) = \langle \phi_m, \phi_{m'} \rangle_{\mathcal{H}}, \tag{17}$$
$$[\mathbf{K_{vu}}]_{km} = \mathrm{Cov}\left(v_k, u_m\right) = \langle \psi_k, \phi_m \rangle_{\mathcal{H}}, \tag{18}$$
$$[\mathbf{K_{vv}}]_{kk'} = \mathrm{Cov}\left(v_k, v_{k'}\right) = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}}. \tag{19}$$

In general, we have

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{\substack{m=1, \\ m'=1}}^{M} \left[\mathbf{K_{uu}^{-1}}\right]_{mm'} \langle \psi_k, \phi_m \rangle_{\mathcal{H}} \langle \phi_{m'}, k(\mathbf{x}_n, \cdot) \rangle_{\mathcal{H}}, \tag{20}$$

and

$$[\mathbf{Q_{vv}}]_{kk'} = \sum_{\substack{m=1, \\ m'=1}}^{M} \left[\mathbf{K_{uu}^{-1}}\right]_{mm'} \langle \psi_k, \phi_m \rangle_{\mathcal{H}} \langle \phi_{m'}, \psi_{k'} \rangle_{\mathcal{H}} \tag{21}$$

We enumerate the combinations in turn and discuss their benefits and drawbacks.

A.1. Base Inducing Variables as Standard Inducing Points

In this case, the base inducing variables are simply standard inducing points $u_m = f(\mathbf{z}_m)$, which is equivalent to defining $\phi_m : \mathbf{x} \mapsto k(\mathbf{z}_m, \mathbf{x})$. Therefore, we have

$$[\mathbf{K_{uf}}]_{mn} = \phi_m(\mathbf{x}_n) = k(\mathbf{z}_m, \mathbf{x}_n), \tag{22}$$
$$[\mathbf{K_{uu}}]_{mm'} = \langle \phi_m, \phi_{m'} \rangle_{\mathcal{H}} = \langle k(\mathbf{z}_m, \cdot), k(\mathbf{z}_{m'}, \cdot) \rangle_{\mathcal{H}} = k(\mathbf{z}_m, \mathbf{z}_{m'}). \tag{23}$$

A.1.1. Orthogonal Inducing Variables as Standard Inducing Points

In this case, the orthogonal inducing variables are simply standard inducing points $v_k = f(\mathbf{w}_m)$, which is equivalent to defining $\psi_k : \mathbf{x} \mapsto k(\mathbf{w}_k, \mathbf{x})$. Note that this combination is precisely the original method of Shi et al. (2020). Hence, we have

$$[\mathbf{K_{vf}}]_{kn} = \psi_k(\mathbf{x}_n) = k(\mathbf{w}_k, \mathbf{x}_n), \tag{24}$$

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \langle k(\mathbf{w}_k, \cdot), k(\mathbf{z}_m, \cdot) \rangle_{\mathcal{H}} = k(\mathbf{w}_k, \mathbf{z}_m), \tag{25}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \langle k(\mathbf{w}_k, \cdot), k(\mathbf{w}_{k'}, \cdot) \rangle_{\mathcal{H}} = k(\mathbf{w}_k, \mathbf{w}_{k'}). \tag{26}$$

This leads to

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \sum_{m'=1}^{M} \left[\mathbf{K_{uu}^{-1}} \right]_{mm'} k(\mathbf{w}_k, \mathbf{z}_m) k(\mathbf{z}_{m'}, \mathbf{x}_n), \tag{27}$$

and $\mathbf{S_{vf}}$ does not simplify further.

A.1.2. Orthogonal Inducing Variables as Spherical Harmonics

Here, orthogonal inducing variables are defined through inter-domain features $v_k = \langle f, \psi_k \rangle_{\mathcal{H}}$ where $\psi_k \triangleq Y_{\ell,j}$, the spherical harmonic of level ℓ and order j, and k indexes the pairs (ℓ, j). Hence, $\mathbf{K_{vf}}$ is as expressed in eq. 16,

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \langle \psi_k, k(\mathbf{z}_m, \cdot) \rangle_{\mathcal{H}} = \psi_k(\mathbf{z}_m) \tag{28}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \lambda_k^{-1} \delta_{kk'} \tag{29}$$

This leads to

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \sum_{m'=1}^{M} \left[\mathbf{K_{uu}^{-1}} \right]_{mm'} \psi_k(\mathbf{z}_m) k(\mathbf{z}_{m'}, \mathbf{x}_n) \tag{30}$$

and $\mathbf{S_{vf}}$ does not simplify further.

A.1.3. Orthogonal Inducing Variables as Spherical NN Activations

Here, orthogonal inducing variables are defined through inter-domain features $v_k = \langle f, \psi_k \rangle_{\mathcal{H}}$ where $\psi_k = H_k$, the k-th unit of a spherical NN activation layer (eq. 6). Again, $\mathbf{K_{vf}}$ is as expressed in eq. 16, and we still get $[\mathbf{K_{vu}}]_{km} = \psi_k(\mathbf{z}_m)$ as in eq. 28 from the preceding case, but we now have

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \sum_{\substack{\ell=0: \\ \lambda_\ell \neq 0}}^{\infty} \frac{\varsigma_\ell^2}{\lambda_\ell} \frac{\ell + \alpha}{\alpha} C_\ell^{(\alpha)} \left(\frac{\mathbf{w}_k^\top \mathbf{w}_{k'}}{\|\mathbf{w}_k\| \|\mathbf{w}_{k'}\|} \right), \tag{31}$$

which leads to

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \sum_{m'=1}^{M} \left[\mathbf{K_{uu}^{-1}} \right]_{mm'} \psi_k(\mathbf{z}_m) k(\mathbf{z}_{m'}, \mathbf{x}_n). \tag{32}$$

and $\mathbf{S_{vf}}$ does not simplify further.

A.2. Base Inducing Variables as Spherical Harmonics

Here, the base inducing variables are defined through inter-domain features $u_m = \langle f, \phi_m \rangle_{\mathcal{H}}$ where $\phi_m \triangleq Y_{\ell,j}$, the spherical harmonic of level ℓ and order j, and m indexes the pairs (ℓ, j). In this case, $[\mathbf{K_{uf}}]_{mn} = \phi_m(\mathbf{x}_n)$ as in eq. 15, and

$$[\mathbf{K_{uu}}]_{mm'} = \langle \phi_m, \phi_{m'} \rangle_{\mathcal{H}} = \lambda_m^{-1} \delta_{mm'}. \tag{33}$$

Therefore, we can already simplify $\mathbf{Q_{vf}}$ from eq. 20 as

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \sum_{m'=1}^{M} \delta_{mm'} \lambda_m \langle \psi_k, \phi_m \rangle_{\mathcal{H}} \phi_{m'}(\mathbf{x}_n) = \sum_{m=1}^{M} \lambda_m \langle \psi_k, \phi_m \rangle_{\mathcal{H}} \phi_m(\mathbf{x}_n) \tag{34}$$

A.2.1. ORTHOGONAL INDUCING VARIABLES AS STANDARD INDUCING POINTS

In this case, the orthogonal inducing variables are simply standard inducing points $v_k = f(\mathbf{w}_m)$, which is equivalent to defining $\psi_k : \mathbf{x} \mapsto k(\mathbf{w}_k, \mathbf{x})$. Hence, we have

$$[\mathbf{K_{vf}}]_{kn} = \psi_k(\mathbf{x}_n) = k(\mathbf{w}_k, \mathbf{x}_n) \tag{35}$$

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \langle k(\mathbf{w}_k, \cdot), \phi_m \rangle_{\mathcal{H}} = \phi_m(\mathbf{w}_k) \tag{36}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \langle k(\mathbf{w}_k, \cdot), k(\mathbf{w}_{k'}, \cdot) \rangle_{\mathcal{H}} = k(\mathbf{w}_k, \mathbf{w}_{k'}) \tag{37}$$

Carrying on from eq. 34,

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \lambda_m \phi_m(\mathbf{w}_k) \phi_m(\mathbf{x}_n) = \sum_{\ell=1}^{L} \sum_{j=1}^{J_\ell^d} \lambda_{\ell,j} Y_{\ell,j}(\mathbf{w}_k) Y_{\ell,j}(\mathbf{x}_n). \tag{38}$$

Now, as $M \to \infty$ (more precisely, as $L \to \infty$), $[\mathbf{Q_{vf}}]_{kn}$ converges to $k(\mathbf{w}_k, \mathbf{x}_n) = [\mathbf{K_{vf}}]_{kn}$. In other words, $[\mathbf{S_{vf}}]_{kn}$ approaches zero in the limit, and the orthogonal component of the variational parameterization vanishes, effectively reducing this combination to standard SVGP. However, for finite M, particularly the modest values that are feasible in practice, the orthogonal component remains and will strictly improve the modelling capacity by capturing the residues that the base component cannot.

A.2.2. ORTHOGONAL INDUCING VARIABLES AS SPHERICAL HARMONICS

Here, orthogonal inducing variables are defined through inter-domain features $v_k = \langle f, \psi_k \rangle_{\mathcal{H}}$ where $\psi_k \triangleq Y_{\ell,j}$, the spherical harmonic of level ℓ and order j, and k indexes the pairs (ℓ, j). Hence, $\mathbf{K_{vf}}$ is as expressed in eq. 16, and

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \lambda_m^{-1} \delta_{mk} \tag{39}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \lambda_k^{-1} \delta_{kk'} \tag{40}$$

This leads to

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \lambda_m \langle \psi_k, \phi_m \rangle_{\mathcal{H}} \phi_m(\mathbf{x}_n) = \sum_{m=1}^{M} \delta_{mk} \phi_m(\mathbf{x}_n) = \phi_k(\mathbf{x}_n) \tag{41}$$

in this case, since $\psi_k = \phi_k$, we have $[\mathbf{S_{vf}}]_{kn} = 0$ and this approach collapses to standard SVGP, regardless of the value of M.

A.2.3. ORTHOGONAL INDUCING VARIABLES AS SPHERICAL NN ACTIVATIONS

Here, orthogonal inducing variables are defined through inter-domain features $v_k = \langle f, \psi_k \rangle_{\mathcal{H}}$ where $\psi_k = H_k$, the k-th unit of a spherical NN activation layer (eq. 6). Again, $\mathbf{K_{vf}}$ is as expressed in eq. 16, and we now have

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \lambda_m^{-1} \varsigma_m \phi_m(\mathbf{w}_k) \tag{42}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \sum_{\substack{\ell=0: \\ \lambda_\ell \neq 0}}^{\infty} \frac{\varsigma_\ell^2}{\lambda_\ell} \frac{\ell + \alpha}{\alpha} C_\ell^{(\alpha)} \left(\frac{\mathbf{w}_k^\top \mathbf{w}_{k'}}{\|\mathbf{w}_k\| \|\mathbf{w}_{k'}\|} \right) \tag{43}$$

which leads to

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \varsigma_m \phi_m(\mathbf{w}_k) \phi_m(\mathbf{x}_n) \tag{44}$$

Similar to Appendix A.2.1, as $M \to \infty$ (more precisely, as $L \to \infty$), $[\mathbf{Q_{vf}}]_{kn}$ converges to $\psi_k(\mathbf{x}_n) = [\mathbf{K_{vf}}]_{kn}$. So again, $[\mathbf{S_{vf}}]_{kn}$ approaches zero in the limit, and the orthogonal component of the variational parameterization vanishes, effectively reducing this combination to standard SVGP. However, for finite M, particularly the small values that are feasible in practice, the orthogonal component remains and will strictly improve the modelling capacity by capturing the residues that the base component cannot.

A.3. Base Inducing Variables as Spherical NN Activations

Here, the base inducing variables are defined through inter-domain features $u_m = \langle f, \phi_m \rangle_{\mathcal{H}}$ where $\phi_m = H_m$, the m-th unit of a spherical NN activation layer (eq. 6).

In this case, $[\mathbf{K_{uf}}]_{mn} = \phi_m(\mathbf{x}_n)$ as in eq. 15, and

$$[\mathbf{K_{uu}}]_{mm'} = \langle \phi_m, \phi_{m'} \rangle_{\mathcal{H}} = \sum_{\ell=0:\lambda_\ell \neq 0}^{\infty} \frac{\varsigma_\ell^2}{\lambda_\ell} \frac{\ell + \alpha}{\alpha} C_\ell^{(\alpha)} \left(\frac{\mathbf{z}_m^\top \mathbf{z}_{m'}}{\|\mathbf{z}_m\| \|\mathbf{z}_{m'}\|} \right) \tag{45}$$

A.3.1. ORTHOGONAL INDUCING VARIABLES AS STANDARD INDUCING POINTS

In this case, the orthogonal inducing variables are simply standard inducing points $v_k = f(\mathbf{w}_m)$, which is equivalent to defining $\psi_k : \mathbf{x} \mapsto k(\mathbf{w}_k, \mathbf{x})$. Hence, we have

$$[\mathbf{K_{vf}}]_{kn} = \psi_k(\mathbf{x}_n) = k(\mathbf{w}_k, \mathbf{x}_n) \tag{46}$$

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \langle k(\mathbf{w}_k, \cdot), \phi_m \rangle_{\mathcal{H}} = \phi_m(\mathbf{w}_k) \tag{47}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \langle k(\mathbf{w}_k, \cdot), k(\mathbf{w}_{k'}, \cdot) \rangle_{\mathcal{H}} = k(\mathbf{w}_k, \mathbf{w}_{k'}) \tag{48}$$

as in Appendix A.2.1.

A.3.2. ORTHOGONAL INDUCING VARIABLES AS SPHERICAL HARMONICS

Here, orthogonal inducing variables are defined through inter-domain features $v_k = \langle f, \psi_k \rangle_{\mathcal{H}}$ where $\psi_k = H_k$, the k-th unit of a spherical NN activation layer (eq. 6).

We have

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \lambda_k^{-1} \varsigma_k \psi_k(\mathbf{z}_m) \tag{49}$$

$$[\mathbf{K_{vf}}]_{kn} = \langle \psi_k, f(\mathbf{x}_n) \rangle_{\mathcal{H}} = \langle \psi_k, k(\mathbf{x}_n, \cdot) \rangle_{\mathcal{H}} = \psi_k(\mathbf{x}_n) \tag{50}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \lambda_k^{-1} \delta_{kk'} \tag{51}$$

This leads to

$$[\mathbf{Q_{vf}}]_{kn} = \sum_{m=1}^{M} \sum_{m'=1}^{M} \left[\mathbf{K_{uu}^{-1}} \right]_{mm'} \lambda_k^{-1} \varsigma_k \psi_k(\mathbf{z}_m) \phi_{m'}(\mathbf{x}_n) \tag{52}$$

A.3.3. ORTHOGONAL INDUCING VARIABLES AS SPHERICAL NN ACTIVATIONS

Here, orthogonal inducing variables are defined through inter-domain features $v_k = \langle f, \psi_k \rangle_{\mathcal{H}}$ where $\psi_k = H_k$, the k-th unit of a spherical NN activation layer (eq. 6).

In this case, we still get

$$[\mathbf{K_{vu}}]_{km} = \langle \psi_k, \phi_m \rangle_{\mathcal{H}} = \sum_{\ell=0:\lambda_\ell \neq 0}^{\infty} \frac{\varsigma_\ell^2}{\lambda_\ell} \frac{\ell + \alpha}{\alpha} C_\ell^{(\alpha)} \left(\frac{\mathbf{w}_k^\top \mathbf{z}_m}{\|\mathbf{w}_k\| \|\mathbf{z}_m\|} \right) \tag{53}$$

$$[\mathbf{K_{vf}}]_{kn} = \langle \psi_k, f(\mathbf{x}_n) \rangle_{\mathcal{H}} = \langle \psi_k, k(\mathbf{x}_n, \cdot) \rangle_{\mathcal{H}} = \psi_k(\mathbf{x}_n) \tag{54}$$

$$[\mathbf{K_{vv}}]_{kk'} = \langle \psi_k, \psi_{k'} \rangle_{\mathcal{H}} = \sum_{\ell=0:\lambda_\ell \neq 0}^{\infty} \frac{\varsigma_\ell^2}{\lambda_\ell} \frac{\ell + \alpha}{\alpha} C_\ell^{(\alpha)} \left(\frac{\mathbf{w}_k^\top \mathbf{w}_{k'}}{\|\mathbf{w}_k\| \|\mathbf{w}_{k'}\|} \right) \tag{55}$$

B. Experimental Set-up and Implementation Details

B.1. Hardware

All experiments were carried out on a consumer-grade laptop computer with an Intel Core™ i7-11800H (8 Cores) @ 4.6GHz Processor, 16GB Memory, and a NVIDIA GeForce RTX™ 3070 Laptop (Mobile/Max-Q) Graphics Card.

B.2. Software

Our method is implemented by extending functionality from the GPFlow software library (Matthews et al., 2017). The code will be released as open-source software upon publication. Additional software dependencies upon which our implementation relies, either directly or indirectly, are enumerated in Table 2.

Table 2. Key software dependencies.

Method	Software Library	URL (github.com/*)
SVGP (Titsias, 2009)	GPFlow	GPflow/GPflow
ODVGP (Salimbeni et al., 2018)	-	hughsalimbeni/orth_decoupled_var_gps
SOLVEGP (Shi et al., 2020)	-	thjashin/solvegp
VISH (Dutordoir et al., 2020)	Spherical Harmonics	vdutor/SphericalHarmonics
ACTIVATED SVGP (Dutordoir et al., 2021)	-	vdutor/ActivatedDeepGPs
-	Bayesian Benchmarks	hughsalimbeni/bayesian_benchmarks

B.3. Hyperparameters

We adopt sensible defaults across all problems and datasets; no hand-tuning is applied to any specific one. The choices of the hyperparameters and other relevant dependencies are summarized as follows:

Optimization. We use the L-BFGS optimizer (Byrd et al., 1995; Zhu et al., 1997) with the default settings from `scipy.optimize` (Virtanen et al., 2020).

Likelihood. The Gaussian likelihood variance is initialized to 1.0 across all experiments.

Kernel parameter initialization. All stationary kernels are initialized with unit lengthscale and amplitude.

Variational parameter initialization. The variational distributions $q(\mathbf{u})$, $q(\mathbf{v}')$ are initialized with zero mean and identity covariance $\mathbf{m} = \mathbf{0}, \mathbf{C} = \mathbf{I}$.

Whitened parameterization. We do *not* use the whitened parameterization (as used, for example, by Murray & Adams (2010); Hensman et al. (2015b)) in either $q(\mathbf{u})$ or $q(\mathbf{v}')$.

Inducing point initialization. We make our best effort to ensure a fair comparison against baselines involving standard inducing points. To this end, we adopt the best practice of first optimizing the variational parameters, not least the inducing input locations $\mathbf{Z}$ (and $\mathbf{W}$ where applicable), before jointly optimizing all of the free parameters. This initialization phase is done for up to 100 iterations of the L-BFGS algorithm.

C. Additional Results

C.1. Regression on Airline Delays Dataset

We repeat the experiment outlined in § 4.3, focusing on a reduced subset of the 2008 U.S. airline delays dataset that consists of 100K randomly selected observations. Unlike the previous experimental set-up, the parameters are optimised for a total of 1,000 epochs. Additionally, we report aggregated results from 5 repetitions across $1/3$ held-out test sets. The results are shown in Figure 10.
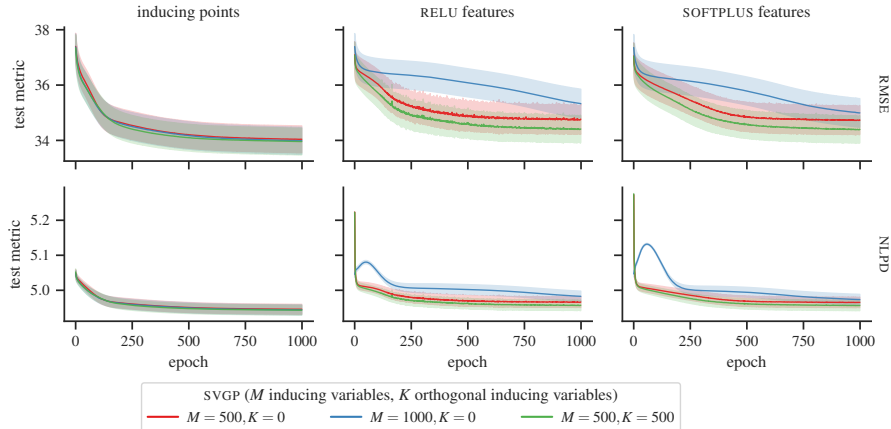

Figure 10. Test metrics, RMSE and NLPD, aggregated across 5 random sub-sampling test splits on a 100K subset of the 2008 U.S. airline delays dataset. Results are shown for models using the *Arccos* kernel with standard inducing points and various activation features. Along the column labeled "inducing points", the red and blue lines (— and —) represent the mini-batch SVGP (Hensman et al., 2013), while the green line (—) represents SOLVEGP (Shi et al., 2020). Along the column labeled "SOFTPLUS features", the red and blue lines (— and —) represent the ACTIVATED SVGP (Dutordoir et al., 2021), while the green line (—) represents our proposed approach.

C.2. Extra UCI Repository Datasets

Results on a few larger regression datasets from the UCI repository can be found in Figure 11. In this analysis, we adopted the same combination of activation features and sparse GP models as described in § 4.2. However, in contrast to § 4.2, we restrict our focus to the Arccos kernel.

C.3. Numerical Tables

For completeness, we include the numerical values of underlying Figures 7 and 9 from § 4 in Tables 3 and 4 below.

Table 3. Numerical values of the test RMSE corresponding to Figure 7.

| dataset	inducing variable kernel SVGP (M inducing variables, K orthogonal inducing variables)	RELU features Arccos	Matérn-5/2	SOFTPLUS features Arccos	Matérn-5/2	inducing points Arccos	Matérn-5/2
CONCRETE ($N = 927, D = 8$)	$M = 128, K = 0$	6.82 ± 0.30	6.92 ± 0.25	6.69 ± 0.29	6.89 ± 0.26	6.58 ± 0.37	5.94 ± 0.25
	$M = 128, K = 128$	5.93 ± 0.38	5.87 ± 0.30	6.06 ± 0.32	5.91 ± 0.33	6.40 ± 0.40	5.75 ± 0.35
	$M = 256, K = 0$	6.56 ± 0.20	6.87 ± 0.26	6.45 ± 0.24	6.86 ± 0.26	6.40 ± 0.35	5.64 ± 0.29
ENERGY ($N = 691, D = 8$)	$M = 128, K = 0$	1.36 ± 0.30	1.58 ± 0.17	0.95 ± 0.10	1.53 ± 0.16	0.47 ± 0.08	0.47 ± 0.08
	$M = 128, K = 128$	0.47 ± 0.08	0.47 ± 0.08	0.47 ± 0.08	0.47 ± 0.08	0.48 ± 0.08	0.47 ± 0.08
	$M = 256, K = 0$	0.91 ± 0.08	1.51 ± 0.15	0.86 ± 0.08	1.51 ± 0.15	0.47 ± 0.08	0.46 ± 0.08
KIN8NM ($N = 7372, D = 8$)	$M = 128, K = 0$	0.12 ± 0.00	0.10 ± 0.00	0.10 ± 0.00	0.10 ± 0.00	0.10 ± 0.00	0.08 ± 0.00
	$M = 128, K = 128$	0.08 ± 0.00	0.08 ± 0.00	0.08 ± 0.00	0.08 ± 0.00	0.09 ± 0.00	0.08 ± 0.00
	$M = 256, K = 0$	0.10 ± 0.00	0.10 ± 0.00	0.09 ± 0.00	0.10 ± 0.00	0.09 ± 0.00	0.08 ± 0.00
POWER ($N = 8611, D = 4$)	$M = 128, K = 0$	4.20 ± 0.16	4.22 ± 0.15	4.20 ± 0.16	4.22 ± 0.15	3.93 ± 0.18	3.90 ± 0.19
	$M = 128, K = 128$	3.96 ± 0.18	3.96 ± 0.19	3.91 ± 0.19	3.96 ± 0.19	3.85 ± 0.18	3.71 ± 0.19
	$M = 256, K = 0$	4.20 ± 0.16	4.22 ± 0.15	4.20 ± 0.16	4.22 ± 0.15	3.83 ± 0.19	3.68 ± 0.19
YACHT ($N = 277, D = 6$)	$M = 128, K = 0$	1.17 ± 0.36	2.43 ± 0.54	1.16 ± 0.36	2.43 ± 0.54	1.06 ± 0.24	0.34 ± 0.19
	$M = 128, K = 128$	0.59 ± 0.23	0.51 ± 0.25	0.60 ± 0.23	0.49 ± 0.29	1.06 ± 0.23	0.34 ± 0.19
	$M = 256, K = 0$	1.16 ± 0.35	2.43 ± 0.54	1.13 ± 0.36	2.43 ± 0.54	1.07 ± 0.23	0.28 ± 0.14
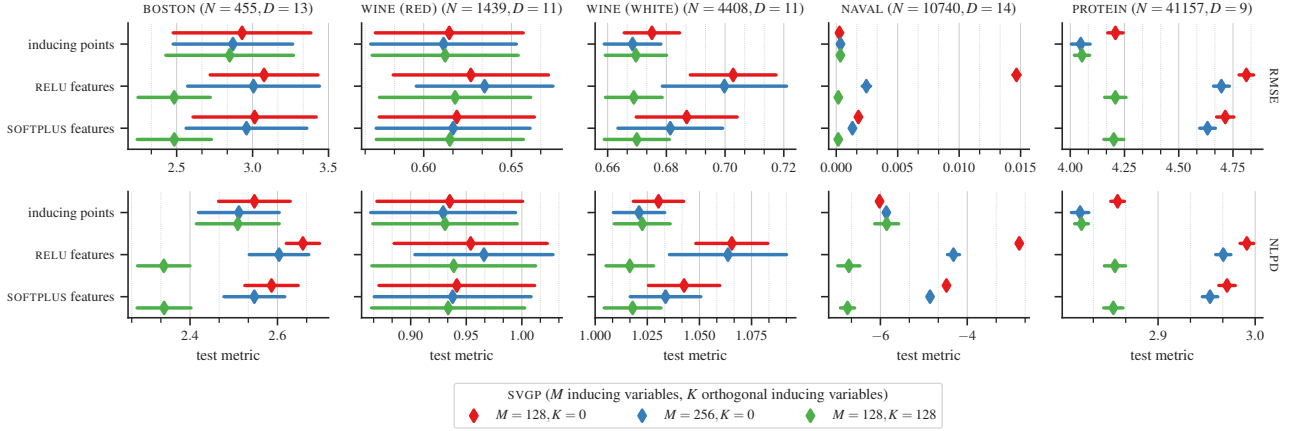
Figure 11. Test metrics, RMSE and NLPD, on an extra set of larger regression problems from the UCI dataset repository for the *Arccos* kernel and various activation features. Along the rows labeled *"inducing points"*, the red and blue markers (◆, ◆) represent the original SVGP model (Titsias, 2009), while the green markers (◆) represent SOLVEGP (Shi et al., 2020). Along the remaining rows, the red and blue markers (◆, ◆) represent the ACTIVATED SVGP (Dutordoir et al., 2021), while the green markers (◆) represent our proposed approach.

Table 4. Numerical values of the test NLPD corresponding to Figure 9.

dataset	inducing variable kernel SVGP (M inducing variables, K orthogonal inducing variables)	RELU features Arccos	RELU features Matérn-5/2	SOFTPLUS features Arccos	SOFTPLUS features Matérn-5/2	inducing points Arccos	inducing points Matérn-5/2
CONCRETE ($N = 927$, $D = 8$)	$M = 128, K = 0$	3.33 ± 0.04	3.36 ± 0.02	3.31 ± 0.04	3.36 ± 0.03	3.30 ± 0.05	3.19 ± 0.07
	$M = 128, K = 128$	3.19 ± 0.08	3.18 ± 0.08	3.22 ± 0.08	3.18 ± 0.08	3.27 ± 0.06	3.12 ± 0.08
	$M = 256, K = 0$	3.29 ± 0.03	3.35 ± 0.03	3.27 ± 0.04	3.35 ± 0.03	3.26 ± 0.05	3.11 ± 0.09
ENERGY ($N = 691$, $D = 8$)	$M = 128, K = 0$	1.83 ± 0.12	2.03 ± 0.04	1.49 ± 0.04	2.01 ± 0.04	0.70 ± 0.17	0.68 ± 0.20
	$M = 128, K = 128$	0.68 ± 0.20	0.68 ± 0.20	0.69 ± 0.20	0.69 ± 0.19	0.70 ± 0.18	0.67 ± 0.19
	$M = 256, K = 0$	1.43 ± 0.03	1.99 ± 0.04	1.38 ± 0.03	1.99 ± 0.04	0.70 ± 0.18	0.67 ± 0.20
KIN8NM ($N = 7372$, $D = 8$)	$M = 128, K = 0$	-0.69 ± 0.02	-0.78 ± 0.02	-0.83 ± 0.02	-0.82 ± 0.02	-0.85 ± 0.02	-1.03 ± 0.02
	$M = 128, K = 128$	-1.03 ± 0.02	-1.03 ± 0.02	-1.03 ± 0.02	-1.03 ± 0.02	-0.90 ± 0.01	-1.10 ± 0.02
	$M = 256, K = 0$	-0.89 ± 0.02	-0.83 ± 0.02	-0.95 ± 0.02	-0.84 ± 0.02	-0.90 ± 0.01	-1.10 ± 0.02
POWER ($N = 8611$, $D = 4$)	$M = 128, K = 0$	2.85 ± 0.04	2.86 ± 0.03	2.85 ± 0.04	2.86 ± 0.03	2.79 ± 0.04	2.78 ± 0.05
	$M = 128, K = 128$	2.80 ± 0.05	2.80 ± 0.05	2.79 ± 0.05	2.80 ± 0.05	2.77 ± 0.04	2.73 ± 0.05
	$M = 256, K = 0$	2.85 ± 0.04	2.86 ± 0.03	2.85 ± 0.04	2.86 ± 0.03	2.77 ± 0.05	2.73 ± 0.05
YACHT ($N = 277$, $D = 6$)	$M = 128, K = 0$	1.81 ± 0.08	2.51 ± 0.06	1.80 ± 0.08	2.51 ± 0.06	1.50 ± 0.15	0.33 ± 0.25
	$M = 128, K = 128$	0.91 ± 0.26	0.73 ± 0.40	0.92 ± 0.26	0.70 ± 0.41	1.51 ± 0.17	0.30 ± 0.23
	$M = 256, K = 0$	1.81 ± 0.08	2.51 ± 0.06	1.79 ± 0.08	2.51 ± 0.06	1.52 ± 0.18	0.07 ± 0.30