Linear Time GPs for Inferring Latent Trajectories from Neural Spike Trains

Matthew Dowling [1] **Yuan Zhao** [2] **Il Memming Park** [3]

Abstract

Latent Gaussian process (GP) models are widely used in neuroscience to uncover hidden state evolutions from sequential observations, mainly in neural activity recordings. While latent GP models provide a principled and powerful solution in theory, the intractable posterior in non-conjugate settings necessitates approximate inference schemes, which may lack scalability. In this work, we propose cvHM, a general inference framework for latent GP models leveraging Hida-Matérn kernels and conjugate computation variational inference (CVI). With cvHM, we are able to perform variational inference of latent neural trajectories with linear time complexity for arbitrary likelihoods. The reparameterization of stationary kernels using Hida-Matérn GPs helps us connect the latent variable models that encode prior assumptions through dynamical systems to those that encode trajectory assumptions through GPs. In contrast to previous work, we use bidirectional information filtering, leading to a more concise implementation. Furthermore, we employ the *Whittle* approximate likelihood to achieve highly efficient hyperparameter learning.

1 Introduction

Arguably the spatiotemporal structure of neural population activity implements neural computation. Although it is not directly observable, recovery of the (effective) latent neural state evolution from recordings of neural activity is possible (Paninski et al., 2010; Kao et al., 2015), and is critical for advancing our understanding of neural computation. Strong experimental evidence supporting the existence of low dimensional neural manifolds has fueled research into developing statistical models that can be used to infer the dynamics underlying neural computation (Macke et al., 2011; Pfau et al., 2013; Archer et al., 2014; Frigola et al., 2014; Pandarinath et al., 2018). These methods usually fall under the header of latent variable models (LVMs) and posit that the observed neural activity can be sufficiently explained by linear or nonlinear mappings of latent dynamics (Pei et al., 2021). Among those, a large class of LVMs employ Gaussian processes (GPs) to specify *a priori* beliefs on the temporal structure of latent trajectories (Yu et al., 2009; Zhao & Park, 2017; Koyama et al., 2010; Jensen et al., 2021).

The success and ubiquitous use of GPs is due in part to favorable properties such as universality, flexibility, and intuitive control over smoothness via time/length scale and differentiablility. However, GP inference generally lacks scalability and non-Gaussian observations make the exact posterior intractable. Though approximate methods like sparse GPs (Titsias, 2009) can help, this comes at the price of accuracy and expressiveness. Variational inference is widely used to enable computationally efficient approximations, however, a naive implementation still requires solving a large and dense linear system with time complexity of $\mathcal{O}(T^3)$ and space complexity $\mathcal{O}(T^2)$ for a sequence of length T.

In this work, we combine two recent developments in GP inference, the Hida-Matérn (HM) kernels and conjugate computation variational inference (CVI). The linear state space representation of GP through HM kernels (Dowling et al., 2021) allows for efficient latent trajectory inference via Kalman filtering/smoothing. Utilizing natural gradients of the exponential family, CVI further reduces VI to a numerically elegant iterative optimization (Khan & Lin, 2017). As a result, the **conjugate variational Hida-Matérn GP (cvHM)** framework accelerates latent GP inference to linear time while maintaining flexibility of kernel choice and computationally efficient hyperparameter optimization. Furthermore, we introduce the Whittle (marginal) likelihood as an attractive approximation for GP hyperparameter learning.

Our contributions are the following: **(i)** We propose cvHM, combining Hida-Matérn GPs and CVI , as a tool for extracting latent trajectories from multivariate (neural) time series in linear time complexity. **(ii)** We show that the *information* filter in tandem with CVI results in a more concise

[1] Stony Brook University, New York, USA [2] National Institute of Mental Health, USA [3] Champalimaud Research, Champalimaud Foundation, Portugal. Correspondence to: Matthew Dowling <matthew.dowling@stonybrook.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

inference scheme than the mean/variance Kalman filter; an added bonus of using the information filter is that natural parameters returned from simultaneous forward/backward filters can be combined additively to give us the natural parameters of the posterior. **(iii)** We show that the Whittle likelihood approximation is more sample efficient for hyperparameter optimization and has a favorable time complexity of $\mathcal{O}(LT\log T)$, compared to $\mathcal{O}(TL_S^3)$ of (the lower bound of) the marginal log-likelihood, where $L_S \geq L$.

2 Background

2.1 GP models of latent trajectories

In this work, we are interested LVMs that define a linear/nonlinear mapping (2) between the latent state and observed variables, and impose assumptions on the temporal structure of latent state evolution through a GP prior (1),

$$z_l(t) \sim \mathcal{GP}(m_l(t), k_l(t,t')) \qquad \text{(latent processes)} \quad (1)$$
$$\mathbf{y}_t \mid \mathbf{z}_t \sim P\left(\mathbf{y} \mid g(\mathbf{z}_t)\right) \qquad \text{(observation model)} \quad (2)$$

where $\mathbf{z}_t = (z_1(t), \dots, z_L(t))^\top$ is one of the L unobserved latent processes modeled by a GP with mean and covariance functions m_l and k_l respectively, and $\mathbf{y}_t \in \mathbb{R}^N$ represents the observation at time t that probabilistically depends on the temporal slice of all latent processes at time t, $\mathbf{z}_t \in \mathbb{R}^L$. Once the data has been observed, the goal of Bayesian inference is to find the posterior distribution of latent processes, $p(\mathbf{z}_{1:L} \mid \mathbf{y}_{1:T})$, as well as the (hyper-)parameters of m_l, k_l and g. Without loss of generality, we follow the standard practice of setting the mean function to be zero.

The linear and Gaussian assumption provides a convenient parameterization for the observation model (e.g. GPFA (Yu et al., 2009))

$$\mathbf{y}_t = \mathbf{C}\mathbf{z}_t + \mathbf{b} + \boldsymbol{\nu}_t \qquad (3)$$

where $\mathbf{C}$ is a readout matrix, $\mathbf{b}$ is a bias, and $\boldsymbol{\nu}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$. The linear Gaussian assumption makes it natural to appeal to Expectation-Maximization (EM) for learning hyperparameters, since the complete data log-likelihood, and posterior can be evaluated in closed form (Yu et al., 2009; Dempster et al., 1977). However, evaluating these closed form expressions to compute the posterior still requires solving a large linear system of equations (Rasmussen & Williams, 2005).

Despite the convenience of a tractable posterior, the linear Gaussian observation model is not always suitable for various types of observations, e.g. point processes. Many methods (Macke et al., 2011; Adam et al., 2016; Zhao & Park, 2017; Pandarinath et al., 2018) thus relax the assumption to non-Gaussian conditional distributions, i.e.

$$y_{n,t} \mid \mathbf{z}_t \sim p\left(y_{n,t} \mid g(\mathbf{z}_t)\right) \qquad (4)$$

where n indexes the observation dimension and g is a generic function. The price to pay for using a non-Gaussian likelihood is an intractable posterior, necessitating the use of approximate Bayesian methods. Variational inference methods combat this by choosing a tractable family of distributions to approximate the posterior, for instance, a factorized Gaussian $q(\mathbf{z}) = \prod_{l=1}^{L} \mathcal{N}(\mathbf{m}_l, \mathbf{P}_l)$, so that the evidence lower bound (ELBO),

$$\mathcal{L} = \mathbb{E}_{q(\mathbf{z})} \log p(\mathbf{y}_{1:T} \mid \mathbf{z}_1, \dots, \mathbf{z}_L) - \mathbb{D}_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z})) \qquad (5)$$

is maximized with respect to the variational parameters $(\mathbf{m}_l, \mathbf{P}_l)_{l=1}^{L}$. However, variational inference suffers from similar scalability issues as exact GP inference; computation of the KL divergence between Gaussian distributions with unstructured covariance matrices scales as $\mathcal{O}(T^3)$.

2.2 Hida-Matérn GPs: A state space view of GP

The state-space model (SSM) representation of stationary GPs, that are finitely differentiable in mean-square, has proven itself as a useful tool for reducing the time complexity of GP posterior inference (Hartikainen & Sarkka, 2010; Chang et al., 2020; Solin et al., 2018). It is easy to construct state-space representations of GPs when their kernel can be written as a linear combination of Hida-Matérn kernels; in that case, the GP in tandem with its mean square derivatives is Markovian, which allows for the use of fast inference routines (Hida & Hitsuda, 1993; Lévy, 1956). Furthermore, the linear combinations of Hida-Matérn kernels can approximate the covariance function of *any* stationary GP (Dowling et al., 2021), making them arbitrarily expressive.

Consider a stationary GP with an M-th order Hida-Matérn kernel,

$$\begin{aligned} \text{cov}(z_t, z_{t+\tau}) &= k_{H,M}(\tau;\, \sigma^2, b, \rho) \\ &= \sigma^2 \cos(2\pi b\tau) k_{\text{Matérn}}(\tau;\, \rho, M + \tfrac{1}{2}) \end{aligned} \qquad (6)$$

where $k_{\text{Matérn}}(\tau;\, \rho, M + \frac{1}{2})$ is the Matérn kernel with lengthscale ρ and smoothness parameter $\nu = M + \frac{1}{2}$ (Rasmussen & Williams, 2005). Such a GP is M times differentiable in the mean-squared sense (Jazwinski, 2007). Even though a mean square differentiable GP *is not* always Markovian, the vector process $\mathbf{z}_t^S = [z_t, z_t^{(1)}, \dots, z_t^{(M)}]$ *is* Markovian, where $z_s^{(i)}$ is the i^{th} mean square derivative of z_s. Since Gaussainity is preserved under linear operations, the mean square derivative of a GP is also a GP, and so it is important to be able to compute the multi-output covariance function between a GP and its mean square derivatives (Álvarez & Lawrence, 2011). Fortunately, computing the multi-output covariance between a GP and its mean square derivatives only requires computing appropriate derivatives of the kernel function as we have the relation that $\text{cov}(z_t^{(i)}, z_{t+\tau}^{(j)}) = (-1)^j k^{(i+j)}(\tau)$ with $k^{(i+j)}(\tau)$, the $(i+j)^{\text{th}}$ derivative of $k(\tau)$ with respect to τ. Thus the joint distribution between any two time points in the vector process is

$$p(\mathbf{z}^S_{t+\tau}, \mathbf{z}^S_t) = \mathcal{N}\left(\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K}(0) & \mathbf{K}(\tau)\\ \mathbf{K}(\tau)^\top & \mathbf{K}(0)\end{bmatrix}\right) \tag{7}$$

where $\left[\mathbf{K}^S(\tau)\right]_{ij} = (-1)^j k^{(i+j)}(\tau)$ is the covariance matrix between the GP and its mean square derivatives τ time units apart. Now, as a result of the Markov property, the joint distribution of a Hida-Matérn GP can be factored as $p(\mathbf{z}_1)\prod p(\mathbf{z}_t \mid \mathbf{z}_{t-1})$; using the marginalization property of Gaussian distributions, we can explicitly describe the generative process underlying the GP as the following linear dynamical system (LDS) (Dowling et al., 2021),

$$\mathbf{z}^S_{t+\tau} = \mathbf{A}(\tau)\mathbf{z}^S_t + \mathbf{Q}(\tau), \quad \mathbf{Q}(\tau) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}(\tau)) \tag{8}$$

where

$$\mathbf{A}(\tau) = \mathbf{K}(\tau)\mathbf{K}(0)^{-1} \tag{9}$$

$$\mathbf{Q}(\tau) = \mathbf{K}(0) - \mathbf{K}(\tau)\mathbf{K}(0)^{-1}\mathbf{K}(\tau)^\top \tag{10}$$

State space representation of latent GP models The covariance of the vector process $\mathbf{z}^S_t$ coincides exactly with the covariance of the multi-output GP defined by the kernel $\mathbf{K}^S(\tau)$. With the representation of stationary and finitely differentiable GPs through the state-space representation of the Hida-Matérn GPs, we can rewrite the generative model defined by Eq. (1) and Eq. (3) as follows

$$\mathbf{z}^S_{l,t+\tau} = \mathbf{A}_l(\tau)\mathbf{z}^S_{l,t} + \boldsymbol{\epsilon}_l(\tau) \tag{11}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{H}\mathbf{z}^S_t + \mathbf{b} + \boldsymbol{\nu}_t \tag{12}$$

where $\mathbf{z}^S_t$ is formed by stacking L vector processes, i.e. $\mathbf{z}^S_t = [\mathbf{z}^S_{1,t}, \ldots, \mathbf{z}^S_{L,t}]$, $\mathbf{H}$ is a $L \times L_S$ selector matrix that extracts $\mathbf{z}_t$ from $\mathbf{z}^S_t$, and $L_S = \sum_{l=1}^L M_l$ is the dimension of the *extended state space* – the dimensionality required to represent all L latent processes in addition to their mean square derivatives. The LDS formulation alleviates unfavorable computational complexity, allows inference through the well known Kalman filter and smoothing algorithms, and allows posterior inference in linear time with respect to sequence length (Anderson & Moore, 1979; Särkkä, 2011). In addition, framing the problem in this manner facilitates the use of tools and techniques from the vast literature of Gaussian linear dynamical systems. Application of the Kalman filter and smoothing algorithms to infer the posterior admit time complexity of $\mathcal{O}(L_S^3 T)$ but since $L_S \ll T$ the price is negligible; in the case of large L_S an asymptotic version of the state-space model can be used to avoid expensive operations in the smoothing step (Solin et al., 2018). Thus, given a stationary GP kernel (component) with exactly M derivatives, there exists an equivalent LDS GP formed by appending $M-1$ extra latent dimensions.

2.3 Conjugate variational inference

Adapting the state-space representation (11), we can rewrite the general form of observation model (4) into

$$\mathbf{y}_t \mid \mathbf{z}_t \sim p\left(\mathbf{y}_t \mid g\left(\mathbf{H}\mathbf{z}^S_t\right)\right) \tag{13}$$

Unfortunately, any non-Gaussian likelihood prohibits the immediate use of Kalman filtering and smoothing, and obscures the path to computationally feasible inference.

Recent works (Hamelijnck et al., 2021; Chang et al., 2020) have demonstrated how conjugate computation variational inference (Khan & Lin, 2017) can be used to exploit the SSM representation of GPs for linear time approximate inference. In a nutshell, CVI takes advantage of the fact that, when the variational approximation and prior are in the same exponential family, one step of natural gradient ascent[1] on the ELBO reduces to a conjugate Bayesian update. Supposing that $q(\mathbf{z} \mid \boldsymbol{\lambda}) \approx p(\mathbf{z} \mid \mathbf{y})$ is the variational posterior with natural parameter $\boldsymbol{\lambda}$ and mean parameter $\boldsymbol{\mu}$, and $p(\mathbf{z} \mid \boldsymbol{\lambda}_0)$ is the prior with natural parameter $\boldsymbol{\lambda}_0$, one natural gradient step on the ELBO with learning rate α is equivalent to

$$q(\mathbf{z} \mid \boldsymbol{\lambda}_{k+1}) \propto \underbrace{\exp\left(\tilde{\boldsymbol{\lambda}}_k^\top \mathbf{T}(\mathbf{z})\right)}_{\propto p(\tilde{\mathbf{y}}_k \mid \mathbf{z})} p(\mathbf{z} \mid \boldsymbol{\lambda}_0) \tag{14}$$

$$\begin{aligned}\tilde{\boldsymbol{\lambda}}_{k+1} &= (1-\alpha_k)\tilde{\boldsymbol{\lambda}}_k \\ &\quad + \alpha_k \sum\nolimits_t \nabla_{\boldsymbol{\mu}_{k+1}} \mathbb{E}_{q(z_t \mid \boldsymbol{\lambda}_{k+1})} \log p(y_t \mid z_t)\end{aligned} \tag{15}$$

where $\tilde{\boldsymbol{\lambda}}$ are auxiliary variables that can be considered natural parameters of *pseudo observations* $\tilde{\mathbf{y}}$, and $p(\tilde{\mathbf{y}}_k \mid \mathbf{z})$ is the exponential family distribution that would have $p(\mathbf{z} \mid \boldsymbol{\lambda}_0)$ as its conjugate prior. A principled initialization for CVI is to set $\boldsymbol{\lambda}_1 = \boldsymbol{\lambda}_0$ and $\alpha_1 = 1$ so that $\tilde{\boldsymbol{\lambda}}_1 = \sum \nabla_{\boldsymbol{\mu}} \mathbb{E}_{p(\mathbf{z} \mid \boldsymbol{\lambda}_0)} \log p(y_t \mid z_t)$. More details on CVI are provided in App. B.

3 cvHM for non-conjugate latent GP models

Combining Hida-Matérn kernels and CVI, we propose conjugate variational Hida-Matérn (cvHM), an efficient learning approach of (non-conjugate) latent GP models.

3.1 Posterior inference

Now, we are ready to formulate a procedure for posterior inference and parameter learning when the generative model of the data is specified according to Eq. 11 and 13. In order to take advantage of CVI, we will consider variational Gaussian approximations, i.e. $q(\mathbf{z}) = \mathcal{N}(\mathbf{m}, \mathbf{P})$, that can be represented in their natural parameterization as

$$q(\mathbf{z}) = \exp\left(\mathbf{z}^\top \mathbf{J}\mathbf{z} + \mathbf{h}^\top \mathbf{z} - \log Z\right) \tag{16}$$

[1] In App. B, we provide necessary details about exponential family distributions and natural gradient descent

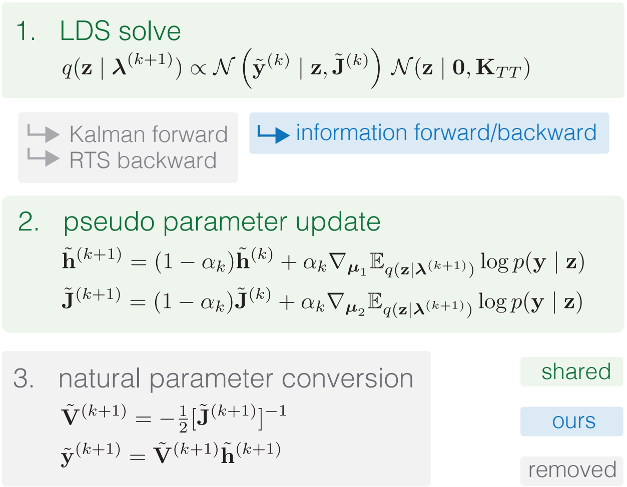
1. LDS solve
$q(\mathbf{z} \mid \boldsymbol{\lambda}^{(k+1)}) \propto \mathcal{N}\left(\tilde{\mathbf{y}}^{(k)} \mid \mathbf{z}, \tilde{\mathbf{J}}^{(k)}\right) \mathcal{N}(\mathbf{z} \mid \mathbf{0}, \mathbf{K}_{TT})$
Kalman forward
RTS backward
information forward/backward
2. pseudo parameter update
$\tilde{\mathbf{h}}^{(k+1)} = (1-\alpha_k)\tilde{\mathbf{h}}^{(k)} + \alpha_k \nabla_{\boldsymbol{\mu}_1} \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\lambda}^{(k+1)})} \log p(\mathbf{y} \mid \mathbf{z})$
$\tilde{\mathbf{J}}^{(k+1)} = (1-\alpha_k)\tilde{\mathbf{J}}^{(k)} + \alpha_k \nabla_{\boldsymbol{\mu}_2} \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\lambda}^{(k+1)})} \log p(\mathbf{y} \mid \mathbf{z})$
3. natural parameter conversion
$\tilde{\mathbf{V}}^{(k+1)} = -\frac{1}{2}[\tilde{\mathbf{J}}^{(k+1)}]^{-1}$
$\tilde{\mathbf{y}}^{(k+1)} = \tilde{\mathbf{V}}^{(k+1)}\tilde{\mathbf{h}}^{(k+1)}$
shared
ours
removed

Figure 1: **Information filtering results in a more concise inference algorithm** By taking advantage of the pseudo observations being in the appropriate form for the information Kalman filter, we can avoid converting from the natural parameter representation. Furthermore, information filtering forward/backward is easily parallelized and more numerically stable than filtering forward/smoothing backward.

with $\mathbf{z} = (z_1, \ldots, z_T)^\top$, normalization constant Z, and natural parameters $\mathbf{J} = -\frac{1}{2}\mathbf{P}^{-1}$, $\mathbf{h} = \mathbf{J}\mathbf{m}$. For conciseness, we will exchangeably use $\boldsymbol{\lambda}$ and $(\mathbf{h}, \mathbf{J})$. A GP prior will have natural parameters $\boldsymbol{\lambda}_0 = (\mathbf{0}, -\frac{1}{2}\mathbf{K}_{TT}^{-1})$ where $\mathbf{K}_{TT}$ is the Grammian evaluated over T points. Since the prior is a GP, the conjugate likelihood is also Gaussian and we can denote the pseudo natural parameters by $(\tilde{\mathbf{h}}, \tilde{\mathbf{J}})$. These parameters can then be converted into Gaussian pseudo observations $\tilde{\mathbf{y}} = \tilde{\mathbf{V}}\tilde{\mathbf{h}}$ with mean $\mathbf{z}$ and covariance $\tilde{\mathbf{V}} = -\frac{1}{2}\tilde{\mathbf{J}}^{-1}$. The first step of a CVI iteration, as in Eq. (14), can now be written in a more familiar form as the following GP regression problem

$$q(\mathbf{z} \mid \boldsymbol{\lambda}_{k+1}) \propto \mathcal{N}(\tilde{\mathbf{y}}_k \mid \mathbf{z}, \tilde{\mathbf{V}}_k) \cdot \mathcal{N}(\mathbf{z} \mid \mathbf{0}, \mathbf{K}_{TT}) \qquad (17)$$

Then the pseudo natural parameters, $\tilde{\mathbf{h}}$ and $\tilde{\mathbf{J}}$, can be updated using the mean parameter gradient as in Eq. (15). This allows each CVI iteration to be done in $\mathcal{O}(TL_S^3)$ time, where the computational bottleneck is solving for the LDS posterior in Eq. (17) Refinement of the variational approximation through additional gradient steps proceeds by recomputing $(\tilde{\mathbf{y}}, \tilde{\mathbf{V}})$, solving the GP regression problem, then updating $(\tilde{\mathbf{h}}, \tilde{\mathbf{J}})$. We summarize this procedure in Alg. 2 in App. E.

Information Filtering Until now, we have glossed over algorithmic details of how the LDS posterior should be computed. With the Gaussian pseudo observation, it is natural to use Kalman filter and RTS smoother to obtain the posterior mean and covariance. However, this requires the conversion from natural parameter space to mean-variance space (Fig. 1.3), $(\tilde{\mathbf{h}}_t, \tilde{\mathbf{J}}_t) \mapsto (\tilde{\mathbf{y}}_t, \tilde{\mathbf{V}}_t)$, every time after updating the pseudo natural parameters according to Eq. (15) (Fig. 1.2). Not only does this conversion introduce additional computation each CVI iteration, it is liable to introduce numerical round off errors; in App. J, we show that sidestepping these conversions in tandem with information filtering, results in improved inference at lower floating point precisions (results were similar using 64 bit floating point).

Fortunately, these conversions can be avoided if instead of computing the posterior through Kalman filtering/RTS smoothing we use the *information* form of the Kalman filter (Anderson & Moore, 1979; Kailath, 1980). In this situation we can think of posterior computation as a black box operation: the standard Kalman filter operates on $(\tilde{\mathbf{y}}_t, \tilde{\mathbf{V}}_t)$, whereas the information form of the Kalman filter operates on $(\tilde{\mathbf{h}}_t, \tilde{\mathbf{J}}_t)$, thus avoiding parameter conversions. How this change makes using CVI together with state-space GP priors more concise compared to other applications of CVI with state-space GPs in the literature, e.g. (Chang et al., 2020; Hamelijnck et al., 2021), is explained in Fig. 1.

Time-reversed dynamics and bidirectional filtering Inference can be further accelerated by deviating from the practice of filtering forward/smoothing backward. A favorable procedure, especially because we are working in the natural parameterization, is to take a message passing approach (Bishop, 2006), and filter forward while filtering backward in parallel, then combine the statistics from each filter to compute the full posterior. In order to perform backward filtering for an LDS, we require the backward representation of the generative process; for Hida-Matérn GPs, the backward dynamics and backward state-noise are

$$\mathbf{A}^b(\tau) = \mathbf{K}(\tau)^\top \mathbf{K}(0)^{-1} \qquad (18)$$
$$\mathbf{Q}^b(\tau) = \mathbf{K}(0) - \mathbf{K}(\tau)^\top \mathbf{K}(0)^{-1}\mathbf{K}(\tau) \qquad (19)$$

which can be used to describe the generative process given by Eq.13 in backwards time,

$$\mathbf{z}_{l,t}^S = \mathbf{A}_l^b(\tau)\mathbf{z}_{l,t+\tau}^S + \boldsymbol{\epsilon}_l^b(\tau) \qquad (20)$$

Backward filtering returns the marginal posterior statistics of $p(\mathbf{z}_t \mid \mathbf{y}_{t:T})$, the filtering distribution at time t given all data after that point (We provide more details about backward filtering in App. C.1). Thanks to the fact that the marginal prior, and backward/forward filtering approximate distributions are Gaussian (i.e. in the exponential family) if we factor the marginal posterior as

$$q(\mathbf{z}_t \mid \mathbf{y}_{1:T}) \propto \frac{\overbrace{q(\mathbf{z}_t \mid \mathbf{y}_{1:t})}^{\mathcal{N}(\mathbf{m}_t^f, \mathbf{P}_t^f)} \overbrace{q(\mathbf{z}_t \mid \mathbf{y}_{t+1:T})}^{\mathcal{N}(\bar{\mathbf{m}}_t^b, \bar{\mathbf{P}}_t^b)}}{\underbrace{p_{\boldsymbol{\theta}}(\mathbf{z}_t)}_{\mathcal{N}(\mathbf{0}, \mathbf{K}(0))}}, \qquad (21)$$

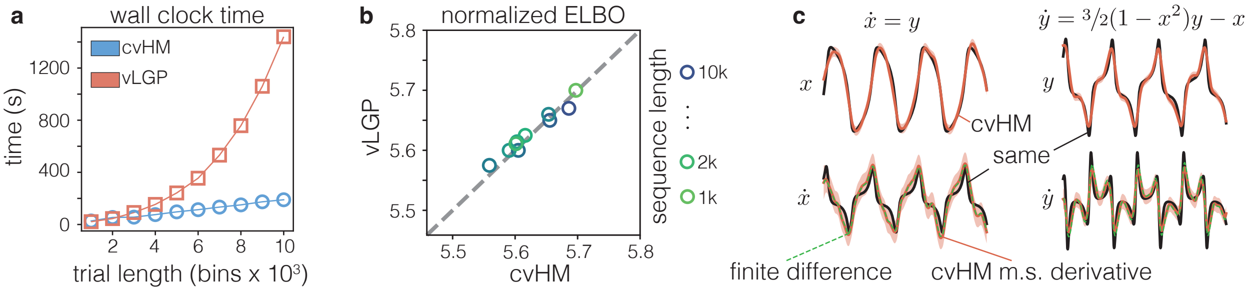

Figure 2: **(a)** cvHM inference scales better than vLGP. Sequence length varies from 1k to 10k in intervals of 1k. Hyperparameters of both methods are kept constant. **(b)** Normalized ELBO (nats/bin) comparison shows same quality inference. **(c)** Van der Pol oscillator experiment. (top) cvHM inference on the 2D latents. (bottom) Time derivative of x and y, and corresponding mean square derivatives inferred by cvHM compared to finite difference of the inferred mean; GP inferred derivatives come at no additional cost and offer calibrated measure of uncertainty that may be useful in latent trajectory analysis to better understand neural computation.

where $q(\mathbf{z}_t \mid \mathbf{y}_{t+1:T}) = \mathbb{E}_{q(\mathbf{z}_{t+1}|\mathbf{y}_{t+1:T})}[p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)]$ then, $q(\mathbf{z}_t \mid \mathbf{y}_{1:T}) = \mathcal{N}(\mathbf{m}_t, \mathbf{P}_t)$, and the posterior marginal statistics can be directly read off

$$\mathbf{P}_t^{-1} = [\mathbf{P}_t^f]^{-1} + [\bar{\mathbf{P}}_t^b]^{-1} - \mathbf{K}(0)^{-1} \tag{22}$$

$$\mathbf{P}_t^{-1}\mathbf{m}_t = [\mathbf{P}_t^f]^{-1}\mathbf{m}_t^f + [\bar{\mathbf{P}}_t^b]^{-1}\bar{\mathbf{m}}_t^b \tag{23}$$

i.e. by adding the natural parameters recovered from the forward/backward filter, and subtracting natural parameters of the prior. Thus, using forward/backward filters make computing the posterior as simple as combining the natural parameters returned from the forwards/backwards information filters. Thanks to stationary assumptions and linearity, the forward and backward filtering are mutually independent, and thus can be performed in parallel – speeding up posterior inference by two-fold.

3.2 Learning GP Hyperparameters

Standard practice for learning hyperparameters of the model is to use variational expectation maximization (VEM) (Turner & Sahani, 2011). For GP regression with a non-conjugate likelihood this would be a computational challenge; every gradient step within the M-step requires recomputing the KL divergence term of the ELBO. A useful result from (Hamelijnck et al., 2021) is that the ELBO can be rewritten by using the GP regression form of the variational approximation to give

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_t \left[\mathbb{E}_{q(\mathbf{z}_t)} \log \frac{p(\mathbf{y}_t \mid \mathbf{z}_t)}{p(\tilde{\mathbf{y}}_t \mid \mathbf{z}_t)} + \log p_{\boldsymbol{\theta}}(\tilde{\mathbf{y}}_t \mid \tilde{\mathbf{y}}_{1:t-1}) \right]$$

where $p_{\boldsymbol{\theta}}(\tilde{\mathbf{y}}_t \mid \tilde{\mathbf{y}}_{1:t-1}) = \mathbb{E}_{\bar{q}_{\boldsymbol{\theta}}(\mathbf{z}_t)}[p(\tilde{\mathbf{y}}_t \mid \mathbf{z}_t)]$ and $\bar{q}_{\boldsymbol{\theta}}(\mathbf{z}_t) = \mathbb{E}_{q(\mathbf{z}_{t-1}|\tilde{\mathbf{y}}_{1:t-1})}[p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{z}_{t-1})]$. Written this way, the ELBO can be evaluated in $\mathcal{O}(TL_S^3)$ time, due to the Markov structure of the variational approximation. The log-marginal likelihood of the auxiliary observations can be computed using the *predicted* values of the Kalman filter used to compute the variational approximation. However, the cubic scaling with L_S could be prohibitive for models with a high-dimensional extended state-space.

Spectral Hyperparameter Optimization Although we can evaluate the ELBO analytically, lets first rewrite it dropping terms independent of $\boldsymbol{\theta}$, so that

$$\mathcal{L}(\boldsymbol{\theta}) = -\mathbb{D}_{\mathrm{KL}}(q(\mathbf{z})||p_{\boldsymbol{\theta}}(\mathbf{z})) = -\mathbb{E}_{q(\mathbf{z})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{z})\right] \tag{24}$$

Now, we will consider a *spectral* approximation[2] of the log-marginal likelihood given via *Whittle's likelihood approximation* (Whittle, 1951; Beran, 2017). Let $S_{\boldsymbol{\theta}}(\omega) = \mathcal{F}[k_{\boldsymbol{\theta}}(\tau)]$ be the power spectral density (PSD) of the prior GP[3], and $Z(\omega) = \mathcal{F}[\mathbf{z}]$. Then, by plugging in Whittle's approximate likelihood we have

$$\log p_{\boldsymbol{\theta}}(\mathbf{z}) = -\tfrac{1}{2}\left(\mathbf{z}^\top \mathbf{K}_{TT}^{-1}\mathbf{z} + \log|\mathbf{K}_{TT}|\right) + C \tag{25}$$

$$\approx -\tfrac{1}{2}\sum_j \left(\log S_{\boldsymbol{\theta}}(\omega_j) + \frac{||Z(\omega_j)||^2}{S_{\boldsymbol{\theta}}(\omega_j)} \right) \tag{26}$$

From here, if we recall that the Fourier transform is a linear transform (e.g. $\mathcal{F}[\mathbf{z}] = \mathbf{F}\mathbf{z}$ where $\mathbf{F} \in \mathbb{R}^{T\times T}$ is the DFT matrix) and Gaussianity is preserved under linear transformations, the expectation can be evaluated so that

$$\mathcal{L}(\boldsymbol{\theta}) \approx -\tfrac{1}{2}\sum_j \left(\log S_{\boldsymbol{\theta}}(\omega_j) + \frac{\mathbb{E}_{q(\mathbf{z})}||\mathbf{f}_j^\top \mathbf{z}||^2}{S_{\boldsymbol{\theta}}(\omega_j)} \right) \tag{27}$$

$$= -\tfrac{1}{2}\sum_j \left(\log S_{\boldsymbol{\theta}}(\omega_j) + \frac{\mathbb{E}_{q(a_j)}\left[a_j^2\right]}{S_{\boldsymbol{\theta}}(\omega_j)} \right) \tag{28}$$

where $q(a_j) = \mathcal{N}(a_j \mid \mathbf{f}_j^\top \mathbf{m}, \mathbf{f}_j^\top \mathbf{P}\mathbf{f}_j)$ and $\mathbf{f}_j$ is the j^{th} row of the DFT matrix, $\mathbf{F}$. Evaluating this bound has an initial

[2] We give an introduction to the Whittle likelihood in App. A

[3] $\mathcal{F}[\cdot]$ is the Fourier transform; $S_{\boldsymbol{\theta}}(\omega)$ & $k_{\boldsymbol{\theta}}(\tau)$ are Fourier duals by the Wiener-Khintchine theorem.

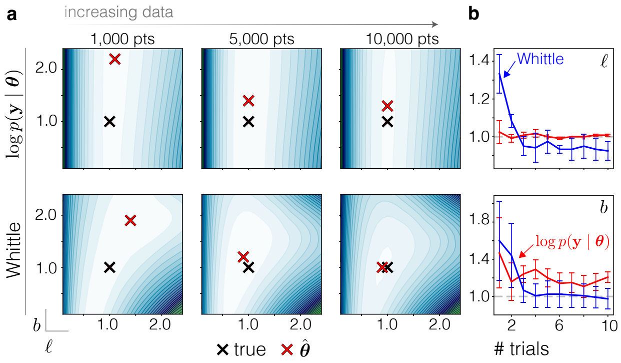

Figure 3: **(a)** We simulate a one-dimensional Gaussian LDS, where the latent variable are sampled from a GP with kernel $k_{H,1}(\tau; 1, 1, 1)$ then compare the log-marginal likelihood and the Whittle approximation as an objective for hyperparameter learning (from left to right 1k/5k/10k observations); $\hat{\boldsymbol{\theta}}$ denotes the parameter that maximizes objective. The log-marginal likelihood does not seem to concentrate well on the frequency parameter. **(b)** Convergence of the log-marginal likelihood and Whittle approximation for hyperparameters averaged over 3 random seeds; the bias for the Whittle approximation is evident, but it also appears to have better convergence properties for b; we apply a Hann taper function before any Fourier transforms taken.

cost of $\mathcal{O}(LT \log T)$ for the Fourier transform of L latent processes, but each gradient step on the hyperparameters is only as costly as evaluating the summation in Eq. (26). The optimal hyperparameters cannot be found in closed form, but the optimal PSD can, and may provide further intuition in regards to using the Whittle likelihood as an objective function. The following proposition, which we derive in the Appendix, states that the PSD maximizing the ELBO at frequency ω_j is $\mathbb{E}_{q(a_j)}\left[a_j^2\right]$.

Proposition 1 (Optimal $S_{\boldsymbol{\theta}}(\omega)$) *The function, $S^*_{\boldsymbol{\theta}}(\omega)$ maximizing the ELBO at frequencies $\omega_1, \ldots, \omega_{T/2}$, is given by*

$$S^*(\omega_j) = \underset{S(\omega_j)}{\operatorname{argmin}}\ \mathcal{L}(S(\omega)) = \mathbb{E}_{q(a_j)}\left[a_j^2\right] \tag{29}$$

This suggests that optimizing the Whittle likelihood attempts to bring the prior PSD closer to the expected periodogram of the posterior latent process. In Fig.3 we compare the Whittle likelihood to the log-marginal likelihood as an objective function. The Whittle likelihood allows us to reduce hyperparameter optimization from $\mathcal{O}(L_S^3 T)$ to $\mathcal{O}(LT \log T)$; furthermore, the Whittle likelihood makes it possible to take advantage of methods/theory from signal processing in a probabilistic context.

4 Related work

Unlike models e.g. (Lawrence, 2005) using GPs to define the functional relationship between latent and observed variables, the LVMs of our interest define a linear/nonlinear mapping between the latent state and observed variables, and use GP to impose *a priori* temporal structure of latent trajectories; P-GPLVM (Wu et al., 2017) considers GP dynamics with a tuning curve function that is also modeled by GP. Markovian linear autoregressive models, such as the Poisson Linear Dynamical System (PLDS), are extended to the nonlinear regime in (Wang et al., 2005; Frigola et al., 2014; Zhao et al., 2022) by modeling the transition function with GP; unlike cvHM which aims to extract smooth latent trajectories, these methods in addition learn the underlying law that governs neural population dynamics. Chang et al. (2020), and Hamelijnck et al. (2021), use CVI and the state-space representation of GPs for non-Gaussian observations, but in the context of spatio-temporal processes without necessarily considering a latent space.

Although the state-space representation of cvHM with Poisson spiking is similar to PLDS (Macke et al., 2011), there are important conceptual differences: PLDS explicitly specifies an LDS that governs the dynamics of the neural state, making it necessary to learn all parameters of the transition matrix. Additionally, although the Laplace approximation used in PLDS is practically convenient, it negates theoretical guarantees of a monotonically increasing marginal likelihood provided by the EM algorithm. In contrast, the LDS in cvHM is determined by the small set of kernel hyperparameters, and is fixed to a degree of mean square differentiability. However, encoding this structure in the prior comes at the cost of increased latent dimensionality and inferring latent variables which do not directly modulate the firing rate. Additionally, while our method applies to any model specifying prior beliefs through GPs finitely differentiable in mean square, such as those in (Loper et al., 2021; Foreman-Mackey et al., 2017; Solin & Särkkä, 2014), computing their state-space representation is not necessarily as straightforward as it is for a Hida-Matérn GP. In App. I, we examine the performance of cvHM on Markovian GP baselines such as those in (Wilkinson et al., 2020).

5 Experiments

In this section, we put cvHM to the test on synthetic data and real neural recordings. First, we verify on toy data that cvHM achieves the same performance as its latent GP model inference counterpart, vLGP (Zhao & Park, 2017), but with linear time complexity. Second, we compare cvHM with GPFA and PLDS on data generated according to dynamics of the Van der Pol system, a non-conservative oscillator with non-linear damping, to show its performance in the case of model mismatch. Third, we apply cvHM on real

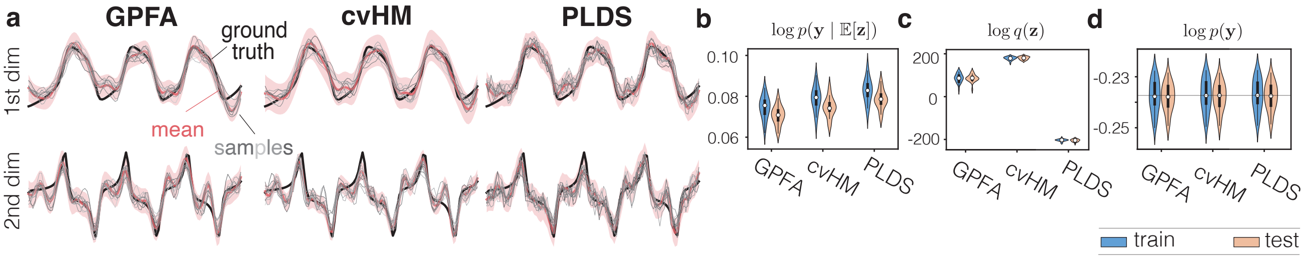

Figure 4: Van der Pol inference comparison of cvHM, GPFA, and PLDS. **(a)** Posterior means, credible intervals, and sample trajectories drawn from each posterior. Note that PLDS samples are much rougher, even though the posterior mean is smooth. **(b)** Reconstruction log-likelihood (bits per spike) based on the posterior mean trajectory (Pei et al., 2021). **(c)** Latent log-likelihood of the ground truth trajectory from the inferred posterior. **(d)** Marginal log-likelihood estimated via sampling. Gray line is centered at cvHM's median test marginal log-likelihood value.

neural recordings[4] taken from a monkey performing a time interval reproduction task. Finally, we demonstrate cvHM's potential impact on experimental design by showcasing its capability to handle long continuous neural recordings[5].

For synthetic data experiments, we use latent trajectories to generate spike trains from 150 neurons through a Poisson generalized linear model (GLM) with the canonical exponential as in Eq. (4). The loading weights, $\mathbf{C}$ and the bias, $\mathbf{b}$, are drawn randomly. We scale these parameters so that all neurons have a realistic baseline firing rate between 5 and 20 Hz. For both real and synthetic neural data, we use 5 ms bins.

5.1 Linear time inference by cvHM

To demonstrate that cvHM achieves the same performance as its counterpart with reduced computation time, we compare against vLGP which uses low-rank Cholesky decomposition to combat the time complexity. In this example, we generated 1D latent trajectories from a Matérn 3/2 GP, with variance and length scale fixed to 1 and 0.01 respectively. To illustrate the reduction in computational complexity, we vary the sequence length from $1,000$ up to $10,000$ bins, and run the experiments on the exact same computing setups to measure wall-clock time. cvHM scales linearly (Fig. 2a) while achieving practically the same performance in terms of ELBO (Fig. 2b).

5.2 Van der Pol oscillator

We compare cvHM against PLDS (Macke et al., 2011) and GPFA (Yu et al., 2009) to quantify how well it performs against models of the same class. For this comparison, we simulate two-dimensional latent trajectories from the classic Van der Pol oscillator (Fig. 2c, top row). Observed spike trains (Fig. 11 in App. D) were generated from the instantaneous latent states ($x(t)$ and $y(t)$). We let all the methods optimize all the hyperparameters.

To quantify the goodness-of-fit for the inferred latent trajectories, we calculate the log-likelihood of the true (simulated) trajectories under the posterior inferred by each method, i.e. $\log q(\mathbf{z}_{1:T})$. The motivation for using this metric is that the reconstruction likelihood measure (Fig. 4b) used in the literature is designed to measure a deterministic firing rate prediction, forcing Bayesian approaches to disregard their posterior distributions. This results in taking the mean latent trajectory of the posterior *before* evaluating the log-likelihood of the observation. In some cases, this may not give good insight into the quality of the inferred latent trajectories. For instance, cvHM finds a higher quality variational posterior than GPFA [6] and PLDS [7] (Fig. 4a,c), while PLDS exhibits a higher reconstruction log-likelihood (Fig. 4a). It is interesting that PLDS tends to trade off smoothness for a better fit to the firing rate since it has a worse posterior over latents. Unfortunately, the latent log-likelihood measure can only be evaluated in simulations where the true latent trajectory is known. Since the time derivative of x is equal to y for the Van der Pol oscillator, we can compare the derivative process inferred by cvHM against its inferred trajectory for y. In Fig. 2c we see that the mean square derivatives on average are similar to the analytic derivatives of the sampled trajectories, while providing a measure of calibrated uncertainty. This illustrates how we can use cvHM to gain additional insights about aspects of latent trajectories that require knowing their time instantaneous derivatives, such as their velocity.

5.3 Electrophysiological Recordings

Time interval reproduction task The utility of any LVM of neural dynamics is the ability to gain qualitative or quantitative understanding about neural computation by examining real data. We use cvHM to examine the

[4] `dandiarchive.org/dandiset/000130`, CC-BY-4.0

[5] `dandiarchive.org/dandiset/000129`, CC-BY-4.0

[6] `https://github.com/NeuralEnsemble/elephant` (Denker et al., 2018), BSD 3 license

[7] `github.com/lindermanlab/ssm`, MIT license

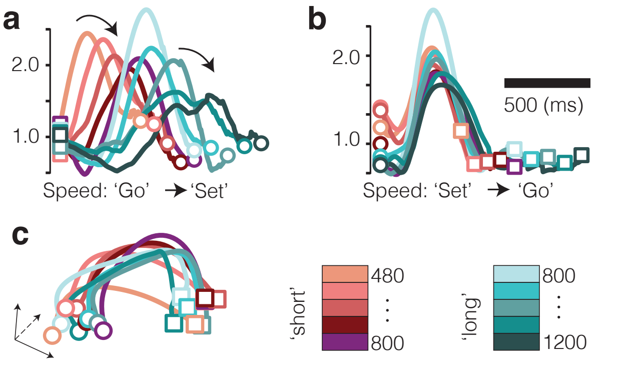

Figure 5: DMFC-RSG dataset 'eye-right' condition. **(a)** Speed of neural trajectories was easily inferred using the mean square derivatives; aligned to 'Go'. Results show that peak speed decreases within increasing intervals under each prior expectation (short vs. long contexts, see (Sohn et al., 2019)). **(b)** Trajectory speeds aligned to 'Set'. **(c)** Latent trajectories.

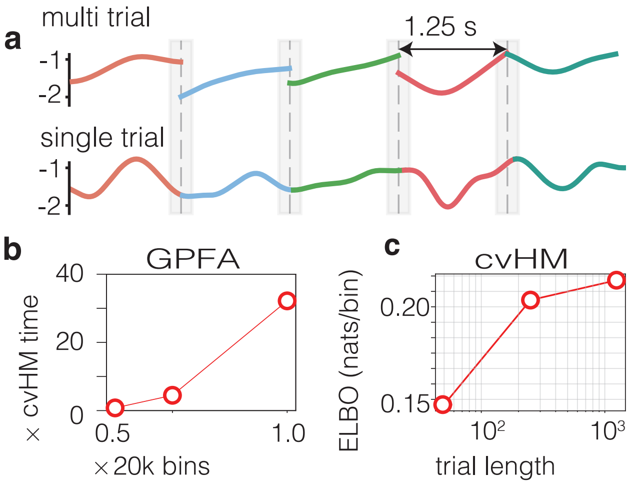

Figure 6: **(a)** Staggered trajectories of a single latent dimension. **(b)** GPFA inference computation time normalized by cvHM inference computation time; for long trials, GPFA becomes infeasible quickly. **(c)** Additionally, decreasing trial length affects the quality of inference as measured by the ELBO; in addition to discontinuities, creating artificial trials affects hyperparameter adversely.

DMFC-RSG dataset (Sohn et al., 2019). The DMFC-RSG dataset includes 54 neurons over 1289 trials recorded from dorso-medial frontal cortex (DMFC), and is known to exhibit low dimensional dynamics. In this task a monkey is exposed to a timing interval demarcated by two visual cues, "Ready" and "Set". Upon seeing the visual cue for "Set" the monkey waits an amount of time and signifies its prediction of the Ready-Set interval (marked "Go"). With cvHM we examine the inferred neural trajectories and their speeds using the mean square derivatives as shown in Fig. 5. We find cvHM verifies the hypothesis that trajectories under the same prior should have speeds that decrease with increasing prediction interval. To validate our results, we also ran GPFA and PLDS on this dataset and found the reconstruction log-likelihood evaluated on the training/test set of both methods to be similar (see Supplement). Again, the utility of cvHM is apparent since to estimate the speed of these trajectories with PLDS and GPFA required using finite difference schemes which do not provide a calibrated measure of uncertainty.

Long trial experiment Many neuroscientific experiments have long trials or no inherent trial structure (O'Doherty et al., 2017). To apply latent GP models, typically one has to split the recording into segments for the sake of computational cost. Practical implementations split trials into even shorter segments for fast EM iterations (e.g. the GPFA implementation of (Denker et al., 2018)[8]). These compromises have drawbacks on inference and hyperparameter tuning. Fortunately, the linear time complexity of cvHM makes it feasible to analyze the continuous recordings in a reasonable amount of time. In this experiment, we demonstrate this and show the effect of trial splitting using the MC-RTT dataset (O'Doherty et al., 2017). This dataset consists of a 15 minute continuous recording from 130 neurons in motor cortex during a reaching task. We use a spike train of 100s (20,000 bins) as a single trial, and split it further into 50s and 25s trials. We fit cvHM to these 3 sets separately.

Figure 6a shows how trial splitting affects the quality of inference. Firstly, the inferred trajectories are staggered at the artificial trial boundaries. Secondly, the inference deteriorates with split (Fig. 6c). Thirdly, the optimal length scale tends to become small as the trials get shorter (see Fig. 12 in the Appendix). To show how inference can quickly become infeasible when using similar GP methods, we compare against GPFA. In order to elucidate the benefit of the state-space approach we normalize sequence length against a 20,000 bin long segment, and wall clock time against cvHM's wall-clock time to infer the posterior of the 20,000 bin long segment (Fig. 6b).

6 Discussion

In this work, we propose cvHM, a latent GP model learning approach combining the recent Hida-Matérn framework and CVI. We showed that cvHM provides competitive inference with favorable linear time complexity for non-conjugate observations. Its high computational efficiency eliminates compromises such as the need to split long trials or use large time bins for practical analyses of long neural recordings, and opens a door to flexible experimental design. Moreover,

[8] https://github.com/NeuralEnsemble/elephant

cvHM provides posterior beliefs about the mean square derivatives of the latent processes as a free lunch, which exhibits potentials in providing further insights for scientific questions. Furthermore, we introduced the Whittle likelihood as an alternative objective for hyperparameter learning of stationary GP models; in spite of bias, the Whittle likelihood approximation could accelerate GP inference as well as in other areas of machine learning.

The GP prior, as we stated early, is essentially a linear dynamical system, so that cvHM, as well as other latent GP models, would mismatch the underlying dynamics if the ground truth is nonlinear. Nonetheless, this does not imply that the inferred trajectories come from a linear DS. Meanwhile, HM theory restricts cvHM to stationary kernels, and could require high numerical precision if high order of derivatives are needed. For practicability, we have found it better to approximate smooth kernels with mixture of low-order HM kernels to avoid ill numerical conditioning.

In future work, we aim to extend cvHM to handle autoregressive observations in order to capture salient details such as neuron refractory periods (Pillow et al., 2008; Zhao & Park, 2017), to utilize nontrivial observation models for nonlinear population coding (e.g. hippocampal CA1 place cells (Wu et al., 2017)), and to account for control inputs in the dynamical system point of view. Moreover, we could employ the sparse GPs (Wilkinson et al., 2021; Adam et al., 2020) to further accelerate cvHM and the state-space representation and iterative nature of CVI may shed on light on online inference.

Acknowledgements

MD and IP were supported by an NSF CAREER Award (IIS-1845836) and NIH RF1DA056404. YZ was supported in part by the National Institute of Mental Health Intramural Research Program (ZIC-MH002968). We thank the anonymous reviewers for their helpful feedback, comments, and motivating insightful experiments.

References

Adam, V., Hensman, J., and Sahani, M. Scalable transformed additive signal decomposition by non-conjugate gaussian process inference. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6. IEEE, September 2016. doi: 10.1109/MLSP.2016.7738855. URL `http://dx.doi.org/10.1109/MLSP.2016.7738855`.

Adam, V., Eleftheriadis, S., Artemev, A., Durrande, N., and Hensman, J. Doubly sparse variational gaussian processes. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2874–2884. PMLR, 26–28 Aug 2020. URL `https://proceedings.mlr.press/v108/adam20a.html`.

Amari, S.-I. Natural gradient works efficiently in learning. *Neural Comput.*, 10(2):251–276, feb 1998. ISSN 0899-7667. doi: 10.1162/089976698300017746. URL `https://doi.org/10.1162/089976698300017746`.

Anderson, B. D. O. and Moore, J. B. *Optimal Filtering*. Prentice-Hall, Englewood Cliffs, N.J., 1979. ISBN 978-0-13-638122-8.

Archer, E. W., Koster, U., Pillow, J. W., and Macke, J. H. Low-dimensional models of neural population activity in sensory cortical circuits. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27*, pp. 343–351. Curran Associates, Inc., 2014.

Beran, J. *Statistics for long-memory processes*. Routledge, 2017.

Bishop, C. M. *Pattern recognition and machine learning (information science and statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

Chang, P. E., Wilkinson, W. J., Khan, M. E., and Solin, A. Fast variational learning in state-space gaussian process models, 2020.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1): 1–38, 1977.

Denker, M., Yegenoglu, A., and Grün, S. Collaborative HPC-enabled workflows on the HBP Collaboratory using the Elephant framework. In *Neuroinformatics 2018*, pp. P19, 2018. doi: 10.12751/incf.ni2018.0019. URL `https://abstracts.g-node.org/conference/NI2018/abstracts#/uuid/023bec4e-0c35-4563-81ce-2c6fac282abd`.

Doob. *Stochastic processes*. John Wiley and Sons, January 1990. ISBN 978-0-471-52369-7.

Dowling, M., Sokół, P., and Park, I. M. Hida-Matérn kernel. 2021. doi: 10.48550/ARXIV.2107.07098. URL `https://arxiv.org/abs/2107.07098`.

Foreman-Mackey, D., Agol, E., Angus, R., and Ambikasaran, S. Fast and scalable gaussian process modeling with applications to astronomical time series. *The Astronomical Journal*, 154, 03 2017. doi: 10.3847/1538-3881/aa9332.

Frigola, R., Chen, Y., and Rasmussen, C. Variational gaussian process State-Space models. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27*, pp. 3680–3688. Curran Associates, Inc., 2014.

Grenander, U. Abstract inference. 1981.

Grenander, U. and Szegö, G. *Toeplitz forms and their applications*. Univ of California Press, 1958.

Hamelijnck, O., Wilkinson, W., Loppi, N., Solin, A., and Damoulas, T. Spatio-temporal variational Gaussian processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Hartikainen, J. and Sarkka, S. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *2010 IEEE International Workshop on Machine Learning for Signal Processing*, pp. 379–384. IEEE, 2010. ISBN 978-1-4244-7875-0. doi: 10.1109/MLSP.2010.5589113. URL http://ieeexplore.ieee.org/document/5589113/.

Hida, T. and Hitsuda, M. *Gaussian Processes*. American Mathematical Society, 1993.

Jazwinski, A. H. *Stochastic Processes and Filtering Theory*. Courier Corporation, January 2007. ISBN 9780486462745. URL https://play.google.com/store/books/details?id=4AqL3vE2J-sC.

Jensen, K., Kao, T.-C., Stone, J., and Hennequin, G. Scalable bayesian gpfa with automatic relevance determination and discrete noise models. *Advances in Neural Information Processing Systems*, 34, 2021.

Kailath, T. *Linear systems*, volume 156. Prentice-Hall Englewood Cliffs, NJ, 1980.

Kao, J. C., Nuyujukian, P., Ryu, S. I., Churchland, M. M., Cunningham, J. P., and Shenoy, K. V. Single-trial dynamics of motor cortex and their applications to brain-machine interfaces. *Nature Communications*, 6:7759+, July 2015. ISSN 2041-1723. doi: 10.1038/ncomms8759. URL http://dx.doi.org/10.1038/ncomms8759.

Khan, M. and Lin, W. Conjugate-Computation Variational Inference : Converting Variational Inference in Non-Conjugate Models to Inferences in Conjugate Models. In Singh, A. and Zhu, J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 878–887. PMLR, 20–22 Apr 2017. URL https://proceedings.mlr.press/v54/khan17a.html.

Khan, M. E. and Nielsen, D. Fast yet simple natural-gradient descent for variational inference in complex models. 2018. doi: 10.48550/ARXIV.1807.04489. URL https://arxiv.org/abs/1807.04489.

Koyama, S., Pérez-Bolde, L. C. C., Shalizi, C. R. R., and Kass, R. E. Approximate methods for State-Space models. *Journal of the American Statistical Association*, 105(489): 170–180, March 2010. ISSN 0162-1459. doi: 10.1198/jasa.2009.tm08326. URL http://dx.doi.org/10.1198/jasa.2009.tm08326.

Lawrence, N. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *Journal of Machine Learning Research*, 6(60):1783–1816, 2005. URL http://jmlr.org/papers/v6/lawrence05a.html.

Loper, J., Blei, D., Cunningham, J. P., and Paninski, L. Linear-time inference for gaussian processes on one dimension, 2021.

Lévy, P. A special problem of brownian motion, and a general theory of gaussian random functions. In Neyman, J. (ed.), *Contributions to Probability Theory*, pp. 133–176. University of California Press, 1956. ISBN 978-0-520-35067-0. doi: 10.1525/9780520350670-013. URL https://www.degruyter.com/document/doi/10.1525/9780520350670-013/html.

Macke, J. H., Buesing, L., Cunningham, J. P., Yu, B. M., Shenoy, K. V., and Sahani, M. Empirical models of spiking in neural populations. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper/2011/file/7143d7fbadfa4693b9eec507d9d37443-Paper.pdf.

O'Doherty, J. E., Cardoso, M. M. B., Makin, J. G., and Sabes, P. N. Nonhuman primate reaching with multichannel sensorimotor cortex electrophysiology. 2017. doi: http://doi.org/10.5281/zenodo.583331.

Pandarinath, C., O'Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., Trautmann, E. M., Kaufman, M. T., Ryu, S. I., Hochberg, L. R., Henderson, J. M., Shenoy, K. V., Abbott, L. F., and Sussillo, D. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15(10):805–815, September 2018. doi: 10.1038/s41592-018-0109-9.

Paninski, L., Ahmadian, Y., Ferreira, D. G. G., Koyama, S., Rahnama Rad, K., Vidne, M., Vogelstein, J., and

Wu, W. A new look at state-space models for neural data. *Journal of Computational Neuroscience*, 29 (1-2):107–126, August 2010. ISSN 1573-6873. doi: 10.1007/s10827-009-0179-x. URL http://dx.doi.org/10.1007/s10827-009-0179-x.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.

Pei, F., Ye, J., Zoltowski, D. M., Wu, A., Chowdhury, R. H., Sohn, H., O'Doherty, J. E., Shenoy, K. V., Kaufman, M. T., Churchland, M., Jazayeri, M., Miller, L. E., Pillow, J., Park, I. M., Dyer, E. L., and Pandarinath, C. Neural latents benchmark '21: evaluating latent variable models of neural population activity. In *Advances in Neural Information Processing Systems (NeurIPS), Track on Datasets and Benchmarks*, 2021. URL https://arxiv.org/abs/2109.04463.

Pfau, D., Pnevmatikakis, E. A., and Paninski, L. Robust learning of low-dimensional dynamics from large neural ensembles. In Burges, C. J. C., Bottou, L., Welling, M., Ghahramani, Z., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 26*, pp. 2391–2399. 2013.

Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E. J., and Simoncelli, E. P. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, August 2008.

Rao, S. S. and Yang, J. Reconciling the gaussian and whittle likelihood with an application to estimation in the frequency domain. *The Annals of Statistics*, 49(5):2774–2802, 2021.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian processes for machine learning*. Adaptive Computation and Machine Learning. The MIT Press, November 2005. ISBN 9780262182539.

Salimbeni, H., Eleftheriadis, S., and Hensman, J. Natural gradients in practice: non-conjugate variational inference in gaussian process models. In Storkey, A. and Perez-Cruz, F. (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 689–697. PMLR, April 2018. URL https://proceedings.mlr.press/v84/salimbeni18a.html.

Särkkä, S. Linear operators and stochastic partial differential equations in gaussian process regression. In *Artificial Neural Networks and Machine Learning – ICANN 2011*, pp. 151–158. Springer Berlin Heidelberg, 2011. doi: 10.1007/978-3-642-21738-8_20. URL http://dx.doi.org/10.1007/978-3-642-21738-8_20.

Sohn, H., Narain, D., and Jazayeri, N. M. M. Bayesian computation through cortical latent dynamics. *Neuron*, 103(5):934–947, sep 2019. ISSN 0896-6273. doi: https://doi.org/10.1016/j.neuron.2019.06.012. URL https://www.sciencedirect.com/science/article/piiS0896627319305628.

Solin, A. and Särkkä, S. Explicit Link Between Periodic Covariance Functions and State Space Models. In Kaski, S. and Corander, J. (eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 904–912, Reykjavik, Iceland, 22–25 Apr 2014. PMLR. URL https://proceedings.mlr.press/v33/solin14.html.

Solin, A., Hensman, J., and Turner, R. E. Infinite-horizon gaussian processes. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/b865367fc4c0845c0682bd466e6ebf4c-Paper.pdf.

Sykulski, A. M., Olhede, S. C., Guillaumin, A. P., Lilly, J. M., and Early, J. J. The debiased whittle likelihood. *Biometrika*, 106(2):251–266, 2019.

Titsias, M. Variational learning of inducing variables in sparse gaussian processes. In van Dyk, D. and Welling, M. (eds.), *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pp. 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL http://proceedings.mlr.press/v5/titsias09a.html.

Turner, R. E. and Sahani, M. Two problems with variational expectation maximisation for time-series models. In Barber, D., Cemgil, T., and Chiappa, S. (eds.), *Bayesian Time series models*, chapter 5, pp. 109–130. Cambridge University Press, 2011.

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.

Wainwright, M. J. and Jordan, M. I. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008. ISSN 1935-8237. doi: 10.1561/2200000001. URL `http://dx.doi.org/10.1561/2200000001`.

Wang, J., Hertzmann, A., and Fleet, D. J. Gaussian process dynamical models. In Weiss, Y., Schölkopf, B., and Platt, J. (eds.), *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005. URL `https://proceedings.neurips.cc/paper/2005/file/ccd45007df44dd0f12098f486e7e8a0f-Paper.pdf`.

Whittle, P. *Hypothesis testing in time series analysis*, volume 4. Almqvist & Wiksells boktr., 1951.

Wilkinson, W., Chang, P., Andersen, M., and Solin, A. State space expectation propagation: Efficient inference schemes for temporal gaussian processes. In *International Conference on Machine Learning*, pp. 10270–10281. PMLR, 2020.

Wilkinson, W., Solin, A., and Adam, V. Sparse algorithms for markovian gaussian processes. In *International Conference on Artificial Intelligence and Statistics*, pp. 1747–1755. PMLR, 2021.

Wu, A., Roy, N. A., Keeley, S., and Pillow, J. W. Gaussian process based nonlinear latent structure discovery in multivariate spike train data. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/b3b4d2dbedc99fe843fd3dedb02f086f-Paper.pdf`.

Yu, B. M., Cunningham, J. P., Santhanam, G., Ryu, S. I., Shenoy, K. V., and Sahani, M. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *Journal of Neurophysiology*, 102(1):614–635, jul 2009. doi: 10.1152/jn.90941.2008. URL `https://doi.org/10.1152/jn.90941.2008`. PMID: 19357332.

Zhao, Y. and Park, I. M. Variational latent gaussian process for recovering single-trial dynamics from population spike trains. *Neural Computation*, 29(5):1293–1316, may 2017. doi: 10.1162/neco_a_00953.

Zhao, Y., Nassar, J., Jordan, I., Bugallo, M., and Park, I. M. Streaming variational Monte Carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2022. doi: 10.1109/TPAMI.2022.3153225.

Álvarez, M. A. and Lawrence, N. D. Computationally Efficient Convolved Multiple Output Gaussian Processes. *Journal of Machine Learning Research*, 12(41):1459–1500, 2011. URL `http://jmlr.org/papers/v12/alvarez11a.html`.

A The Whittle likelihood

In the proposed inference procedure, the Whittle likelihood is used to reduce the computational complexity of hyperparameter optimization. The Whittle likelihood has been used since its conception (Whittle, 1951) to ameliorate the computational difficulties of evaluating the log-marginal likelihood of a Gaussian process; although this is a commonly used approximation in the stochastic processes literature (Sykulski et al., 2019; Rao & Yang, 2021; Grenander, 1981), to the best of our knowledge this is the first time the Whittle likelihood has been used in the context of machine learning and for approximate Bayesian inference with non-Gaussian observations.

For the sake of accessibility, we will walk through a derivation of the Whittle likelihood; any of the aforementioned references can be consulted for further details. Imagine we regularly sample a GP, $z(t) \sim \mathcal{GP}(0, k(\tau))$, at T time points and collect these observations into the vector $\mathbf{z}_{1:T} = \begin{bmatrix} z_1 & \dots & z_T \end{bmatrix}$. Then, the log-likelihood of this particular sample is

$$\log p(\mathbf{z}_{1:T}) = -\tfrac{1}{2} \left(\mathbf{z}_{1:T}^\top \mathbf{K}_{TT}^{-1} \mathbf{z}_{1:T} + \log |\mathbf{K}_{TT}| + T \log 2\pi \right)$$

Unfortunately, evaluating $\mathbf{K}_{TT}^{-1} \mathbf{z}_{1:T}$ and $\log |\mathbf{K}_{TT}|$ when $\mathbf{K}_{TT}$ is dense scales on the order of $\mathcal{O}(T^3)$. However, because the observations were regularly sampled and the kernel is stationary, $\mathbf{K}_{TT}$ will be a Toeplitz matrix (constant along all diagonals). The only necessary result to derive the Whittle likelihood, is a result from (Grenander & Szegö, 1958) which states that asymptotically, $\mathbf{K}_{TT}$ can be decomposed as

$$\mathbf{K}_{TT} \approx \mathbf{F}^H \mathbf{D} \mathbf{F} \tag{30}$$

where $\mathbf{F}$ is the DFT matrix, $\mathbf{D}_{ii} = 1/S_{\boldsymbol{\theta}}(\omega_i)$, and H is the conjugate transpose. Using the fact that $\mathbf{F}\mathbf{z}_{1:T}$ is the DFT transform of $\mathbf{z}_{1:T}$, we can plug in these expressions to arrive at the Whittle likelihood

$$\log p(\mathbf{z}_{1:T}) \approx -\tfrac{1}{2} \sum_j \left(\log S_{\boldsymbol{\theta}}(\omega_j) + \frac{||Z(\omega_j)||^2}{S_{\boldsymbol{\theta}}(\omega_j)} \right)$$

where $\omega_j = (2\pi j)/(\Delta T)$ with Δ the sampling interval. The Whittle approximation is a biased estimate of the log-marginal likelihood, however, there exist improvements over the original approximation meant to reduce the approximation bias (Sykulski et al., 2019). While this bias is unfavorable, the reduction in time complexity of hyperparameter optimization from $\mathcal{O}(L_S^3 T)$ down to $\mathcal{O}(T \log T)$ is substantial. Additionally, this allows tools/techniques from the signal processing literature for spectral estimation to be used in the context of probabilistic inference (Rao & Yang, 2021). In Fig. 3 while effects of the bias are evident (i.e. estimation of l), the Whittle likelihood is substantially less invariant to changes in b; the elliptical landscape of the log-marginal likelihood, as is known for GPs (Wu et al., 2017; Rasmussen & Williams, 2005), complicates optimization.

A.1 Optimal PSD

Proposition 1 (Optimal $S_{\boldsymbol{\theta}}(\omega)$) *The function, $S_{\boldsymbol{\theta}}^*(\omega)$ maximizing the ELBO at frequencies $\omega_1, \dots, \omega_{T/2}$, is given by*

$$S^*(\omega_j) = \underset{S(\omega_j)}{\operatorname{argmin}} \ \mathcal{L}(S(\omega)) = \mathbb{E}_{q(a_j)} \left[a_j^2 \right] \tag{29}$$

Proof: Since terms couple additively across frequencies, we can just concern ourselves with finding

$$S^*(\omega_j) = \operatorname{argmax} -\tfrac{1}{2} \left[\log S_{\boldsymbol{\theta}}(\omega_j) + \frac{\mathbb{E}_{q(a_j)}[a_j^2]}{S_{\boldsymbol{\theta}}(\omega_j)} \right] \tag{31}$$

Now, we can regard $S_{\boldsymbol{\theta}}(\omega_j)$ as an ordinary variable – setting the usual derivative to 0, we have that

$$\frac{1}{S^*(\omega_j)} - \frac{\mathbb{E}_{q(a_j)}[a_j^2]}{S^*(\omega_j)^2} = 0 \tag{32}$$

$$\implies S^*(\omega_j) = \mathbb{E}_{q(a_j)}[a_j^2] \tag{33}$$

so that the optimal PSD at each ω_j is just the expected magnitude of the squared DFT.

B Conjugate computation variational inference (CVI)

For completeness, we give a brief review of conjugate computation variational inference (CVI), for further details consult (Khan & Lin, 2017; Khan & Nielsen, 2018). CVI is applicable when we have a hierarchical Bayesian graphical model

$$\mathbf{z} \sim p_{\boldsymbol{\theta}}(\mathbf{z}) \tag{34}$$

$$\mathbf{y} \mid \mathbf{z} \sim p(\mathbf{y} \mid \mathbf{z}) \tag{35}$$

and we want to find a variational approximation to the posterior, $q(\mathbf{z}) \approx p(\mathbf{z} \mid \mathbf{y})$. If $q(\mathbf{z})$ and $p_{\boldsymbol{\theta}}(\mathbf{z})$ are chosen to be in the same exponential family of distributions (Wainwright & Jordan, 2008), then $q(\mathbf{z})$, can be factored as follows

$$q(\mathbf{z}) = h(\mathbf{z}) \exp(\boldsymbol{\lambda}^\top \mathbf{T}(\mathbf{z}) - A(\boldsymbol{\lambda})) \tag{36}$$

where $h(\mathbf{z})$ is the base measure, $\boldsymbol{\lambda}$ is the natural parameter, $\mathbf{T}(\mathbf{z})$ is the sufficient statistic, and $A(\boldsymbol{\lambda})$ is the log-partition function (Khan & Lin, 2017). Parameters of the variational approximation are easily found through gradient ascent on the ELBO, i.e.

$$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \alpha \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{\lambda}) \tag{37}$$

However, the computation of the KL term can present a significant computational challenge – scaling on the order of $\mathcal{O}(T^3 L^3)$ for Gaussian distributions with dense and unstructured covariance matrices. Furthermore, gradients of the ELBO may not provide a suitable ascent direction in the space of probability distributions (Salimbeni et al., 2018). Alternatively, the natural gradient can be used to take advantage of the information geometry of exponential family distributions (Amari, 1998). The natural gradients are given by $\mathbf{F}(\boldsymbol{\lambda})^{-1} \nabla_{\boldsymbol{\lambda}} \mathcal{L}$ where $\mathbf{F}(\boldsymbol{\lambda}) = \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\lambda})}[\nabla^2_{\boldsymbol{\lambda}} \log q(\mathbf{z})] = \nabla^2_{\boldsymbol{\lambda}} A(\boldsymbol{\lambda})$ is the Fisher information matrix. The inverse Fisher information matrix prohibits naively employing natural gradient descent for large T. However, the Fisher information matrix can be entirely avoided by considering another parameterization of the variational distribution in terms of its *mean parameters*, defined as the expectation of the sufficient statistic[9]

$$\boldsymbol{\mu}(\boldsymbol{\lambda}) = \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\lambda})}[\mathbf{T}(\mathbf{z})] = \nabla_{\boldsymbol{\lambda}} A(\boldsymbol{\lambda}) \tag{38}$$

where the natural parameters are also a function of the mean parameters[10], i.e., we can write $\boldsymbol{\lambda}(\boldsymbol{\mu})$. By the chain rule, natural gradient (w.r.t. $\boldsymbol{\lambda}$) of ELBO can be simply written as the gradient w.r.t. the corresponding mean parameters:

$$\mathbf{F}(\boldsymbol{\lambda})^{-1} \nabla_{\boldsymbol{\lambda}} \mathcal{L} = \mathbf{F}(\boldsymbol{\lambda})^{-1} \left(\nabla_{\boldsymbol{\lambda}} \boldsymbol{\mu}\right) \nabla_{\boldsymbol{\mu}} \mathcal{L} = \nabla_{\boldsymbol{\mu}} \mathcal{L}. \tag{39}$$

Therefore gradient ascent on the ELBO can be done through updates without Fisher information matrix,

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \alpha_k \mathbf{F}(\boldsymbol{\lambda}_k)^{-1} \nabla_{\boldsymbol{\lambda}_k} \mathcal{L} = \boldsymbol{\lambda}_k + \alpha_k \nabla_{\boldsymbol{\mu}_k} \mathcal{L}. \tag{40}$$

where $\alpha_k > 0$ are step sizes. Using these parameterizations, (Khan & Lin, 2017) showed the Kullback-Leibler (KL) divergence between the prior and posterior simplifies as $\nabla_{\boldsymbol{\mu}} \mathbb{D}_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z})) = \boldsymbol{\lambda}_0 - \boldsymbol{\lambda}$, where $\boldsymbol{\lambda}_0$ are the natural parameters of the prior, $p(\mathbf{z})$. Hence, the natural gradient ascent steps become

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \alpha_k(\boldsymbol{\lambda}_0 - \boldsymbol{\lambda}_k) + \alpha_k \sum_t \nabla_{\boldsymbol{\mu}_k} \mathbb{E}_{q(z_t)} \log p(y_t \mid z_t). \tag{41}$$

Iterative updates can be transformed into the following two step procedure (Khan & Lin, 2017) using an auxiliary variable $\tilde{\boldsymbol{\lambda}}$ such that

$$\boldsymbol{\lambda}_{k+1} = \tilde{\boldsymbol{\lambda}}_k + \boldsymbol{\lambda}_0 \tag{42}$$

$$\tilde{\boldsymbol{\lambda}}_{k+1} = (1 - \alpha_k) \tilde{\boldsymbol{\lambda}}_k + \alpha_k \sum_t \nabla_{\boldsymbol{\mu}_k} \mathbb{E}_{q(z_t)} \log p(y_t \mid z_t) \tag{43}$$

where care should be taken to note the dependence between $\boldsymbol{\mu}_k$ and $\boldsymbol{\lambda}_k$. Note that (42) resembles a Bayesian posterior calculation with a conjugate prior, while (43) updates the natural parameters $\tilde{\boldsymbol{\lambda}}_k$ of the (approximate) likelihood. This would suggest that the first step can be thought of as a Bayesian posterior calculation, where $\tilde{\boldsymbol{\lambda}}$ are associated with *pseudo-observations* $\tilde{\mathbf{y}}$ conjugate to the prior.

[9] The natural parameters should be in the *minimal* exponential family form; here minimal is a technical requirement on the linear independence (always achievable) of sufficient statistics so that there exists a one-to-one mapping from the natural to the mean parameterization.

[10] The log-partition function in the natural parameters and the negative entropy in the mean parameters form a dual (Wainwright & Jordan, 2008).

Algorithm 1 Backward information filter

```
**Input:** $\mathbf{y}_{1:T}$, $\mathbf{V}_{1:T}$, $\mathbf{A}^b$, $\mathbf{Q}^b$, $\mathbf{Q}_0$
initialization:
  $\mathbf{h}_{T+1} \leftarrow \mathbf{0}$
  $\mathbf{J}_{T+1} \leftarrow -\frac{1}{2}\mathbf{Q}_0^{-1}$
  $\tilde{\mathbf{h}}_t \leftarrow \mathbf{C}^\top \mathbf{V}_t^{-1} \mathbf{y}_t \quad t = 1, \ldots, T$
  $\tilde{\mathbf{J}}_t \leftarrow \mathbf{C}^\top \mathbf{V}_t^{-1} \mathbf{C} \quad t = 1, \ldots, T$
**for** $t = T$ **to** $1$ **do**
    *prediction step*
    $\mathbf{L} \leftarrow \mathbf{J}_{t+1} + \mathbf{A}^{b\top}\mathbf{Q}^{-b}\mathbf{A}^b$
    $\bar{\mathbf{h}}_t \leftarrow \mathbf{Q}^{-b}\mathbf{A}^b\mathbf{L}^{-1}\mathbf{h}_{t+1}$
    $\bar{\mathbf{J}}_t \leftarrow \mathbf{Q}^{-b} - \mathbf{Q}^{-b}\mathbf{A}^b\mathbf{L}^{-1}\mathbf{A}^{b\top}\mathbf{Q}^{-b}$
    *update step*
    $\mathbf{J}_t \leftarrow \bar{\mathbf{J}}_t + \tilde{\mathbf{J}}_t$
    $\mathbf{h}_t \leftarrow \bar{\mathbf{J}}_t + \tilde{\mathbf{J}}_t$
**end for**
**Return:** $\mathbf{J}_{1:T}, \mathbf{h}_{1:T}$
```

C Probabilistic filtering for posterior inference

C.1 Backward filtering

Consider the following LDS modeled forward in time,

$$\mathbf{z}_{t+1} = \mathbf{A}\mathbf{z}_t + \boldsymbol{\epsilon}_t \tag{44}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{z}_t + \boldsymbol{\eta}_t \tag{45}$$

In our context, this model encodes our a priori belief about the latent dynamics. Associated with the SSM that runs forward in time, is an SSM that runs backward in time

$$\mathbf{z}_t = \mathbf{A}^b\mathbf{z}_{t+1} + \boldsymbol{\epsilon}_t^b \tag{46}$$

$$\mathbf{y}_t = \mathbf{C}\mathbf{z}_t + \boldsymbol{\eta}_t \tag{47}$$

where $\boldsymbol{\epsilon}_t^b \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}^b(\tau))$ is the backward state-noise and $\mathbf{A}^b$ is the backward-time dynamics. In general the backwards dynamics and state-noise satisfy the following equations (Kailath, 1980)

$$\mathbf{A}_{t+1}^b = \mathbf{S}_t\mathbf{A}^\top\mathbf{S}_t^{-1} \tag{48}$$

$$\mathbf{Q}_{t+1}^b = \mathbf{S}_t - \mathbf{A}_{t+1}^b\mathbf{S}_{t+1}\mathbf{A}_{t+1}^{b\top} \tag{49}$$

where $\mathbf{S}_t = \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{z}_t, \mathbf{z}_{t+1})}\left[\mathbf{z}_t\mathbf{z}_{t+1}^\top\right]$. When the prior dynamics is a Hida-Matérn GP and all data are sampled at intervals of τ, then $\mathbf{S}_t = \mathbf{K}(\tau)$ for all t and the dynamics/state-noise are given by

$$\mathbf{A}^b(\tau) = \mathbf{K}(\tau)^\top\mathbf{K}(0)^{-1} \tag{50}$$

$$\mathbf{Q}^b(\tau) = \mathbf{K}(0) - \mathbf{K}(\tau)^\top\mathbf{K}(0)^{-1}\mathbf{K}(\tau) \tag{51}$$

respectively. Now, any filtering algorithm could be used in order to compute the backwards filtering distribution given by $p(\mathbf{z}_t \mid \mathbf{y}_{t:T})$. For example, if we know $p(\mathbf{z}_{t+1} \mid \mathbf{y}_{t+1:T})$, then the predict and update steps of recursive inference are

$$p(\mathbf{z}_t \mid \mathbf{y}_{t+1:T}) = \mathbb{E}_{p(\mathbf{z}_{t+1} \mid \mathbf{y}_{t+1:T})}\left[p_{\boldsymbol{\theta}}(\mathbf{z}_t \mid \mathbf{z}_{t+1})\right] \tag{52}$$

$$p(\mathbf{z}_t \mid \mathbf{y}_{t:T}) \propto p(\mathbf{y}_t \mid \mathbf{z}_t)\, p(\mathbf{z}_t \mid \mathbf{y}_{t+1:T}) \tag{53}$$

C.2 Combining statistics from forward/backward filters

The dual filtering approach described in the main text is the preferred method for recovering statistics of the LDS posterior compared to more common filter forward/smooth backward algorithms – especially in the context of variational inference where the natural parameterization is often more helpful. More than that, dual filtering approach allows us to combine the natural parameters returned from both filters to easily compute posterior statistics.

From the forward filter, we recover $q(\mathbf{z}_t \mid \mathbf{y}_{1:t})$ for all t, through the natural parameters $(\mathbf{h}_t^f, \mathbf{J}_t^f)$. At the same time, from the backward filter, we recover $q(\mathbf{z}_t \mid \mathbf{y}_{t:T})$ through the natural parameters $(\mathbf{h}_t^b, \mathbf{J}_t^b)$; however, we require the natural parameters, $(\bar{\mathbf{h}}_t^b, \bar{\mathbf{J}}_t^b)$, of the backward predictive distribution, $q(\mathbf{z}_t \mid \mathbf{y}_{t+1:T})$, to compute the posterior. Fortunately, they are just a byproduct of running the backward filter.

$$q(\mathbf{z}_t \mid \mathbf{y}_{1:T}) \propto q(\mathbf{y}_{t+1:T} \mid \mathbf{z}_t) q(\mathbf{z}_t \mid \mathbf{y}_{1:t}) \tag{54}$$

$$\propto \frac{\overbrace{q(\mathbf{z}_t \mid \mathbf{y}_{1:t})}^{\mathcal{N}(\mathbf{m}_t^f, \mathbf{P}_t^f)} \overbrace{q(\mathbf{z}_t \mid \mathbf{y}_{t+1:T})}^{\mathcal{N}(\bar{\mathbf{m}}_t^b, \bar{\mathbf{P}}_t^b)}}{\underbrace{p_{\boldsymbol{\theta}}(\mathbf{z}_t)}_{\mathcal{N}(\mathbf{0}, \mathbf{K}(0))}}, \tag{55}$$

where $q(\mathbf{z}_t \mid \mathbf{y}_{t+1:T}) = \mathbb{E}_{q(\mathbf{z}_{t+1} \mid \mathbf{y}_{t+1:T})}[p_{\boldsymbol{\theta}}(\mathbf{z}_{t+1} \mid \mathbf{z}_t)]$ then, $q(\mathbf{z}_t \mid \mathbf{y}_{1:T}) = \mathcal{N}(\mathbf{m}_t, \mathbf{P}_t)$, and the posterior marginal statistics can be directly read off

$$\mathbf{P}_t^{-1} = [\mathbf{P}_t^f]^{-1} + [\bar{\mathbf{P}}_t^b]^{-1} - \mathbf{K}(0)^{-1} \tag{56}$$

$$\mathbf{P}_t^{-1}\mathbf{m}_t = [\mathbf{P}_t^f]^{-1}\mathbf{m}_t^f + [\bar{\mathbf{P}}_t^b]^{-1}\bar{\mathbf{m}}_t^b \tag{57}$$

or, we can just substitute the natural parameter representation so that

$$\mathbf{J}_t = \mathbf{J}_t^f + \bar{\mathbf{J}}_t^b - \mathbf{K}(0)^{-1} \tag{58}$$

$$\mathbf{h}_t = \mathbf{h}_t^f + \bar{\mathbf{h}}_t^b \tag{59}$$

C.3 Information filter

Whereas, the traditional Kalman filtering algorithm uses recursive updates for the latent state mean/covariance, the information filtering algorithm uses recursive updates for the natural parameters of the latent state (Anderson & Moore, 1979). Recall, Kalman filter updates are given according to

$$\mathbf{P}_t = \bar{\mathbf{P}}_t - \bar{\mathbf{P}}_t\mathbf{C}^\top(\mathbf{C}\bar{\mathbf{P}}_t\mathbf{C}^\top + \mathbf{V}_t)^{-1}\mathbf{C}\bar{\mathbf{P}}_t \tag{60}$$

$$= \left(\bar{\mathbf{P}}_t^{-1} + \mathbf{C}^\top\mathbf{V}_t^{-1}\mathbf{C}\right)^{-1} \tag{61}$$

so that,

$$\mathbf{P}_t^{-1} = \bar{\mathbf{P}}_t^{-1} + \mathbf{C}^\top\mathbf{V}_t^{-1}\mathbf{C} \tag{62}$$

Just as easily, we can plug in the expression for $\bar{\mathbf{P}}_t = \mathbf{A}\mathbf{P}_{t-1}\mathbf{A}^\top + \mathbf{Q}$ – invoking the Woodbury identity again reveals that

$$\bar{\mathbf{P}}_t^{-1} = \mathbf{Q} - \mathbf{Q}\mathbf{A}^\top(\mathbf{A}\mathbf{P}_{t-1}^{-1}\mathbf{A}^\top + \mathbf{Q}^{-1})^{-1}\mathbf{A}\mathbf{Q} \tag{63}$$

Illustrating that it could be advantageous to consider a recursion for $\mathbf{P}_t^{-1}$ instead of $\mathbf{P}_t$ – especially if the latent dimensionality is significantly smaller than the number of neurons (which is often the case for neuroscience experiments). More in depth treatment can be found in standard texts such as (Anderson & Moore, 1979; Kailath, 1980).

D Experimental details

D.1 More comparison to the latent GP counterpart

In the main text, we presented the normalized ELBO on a test set of the toy data generated from GP latents. Here we show BPS (bits-per-spike) and normalized ELBO measures in Fig. 7 to further verify that cvHM performs qualitatively the same as vLGP.

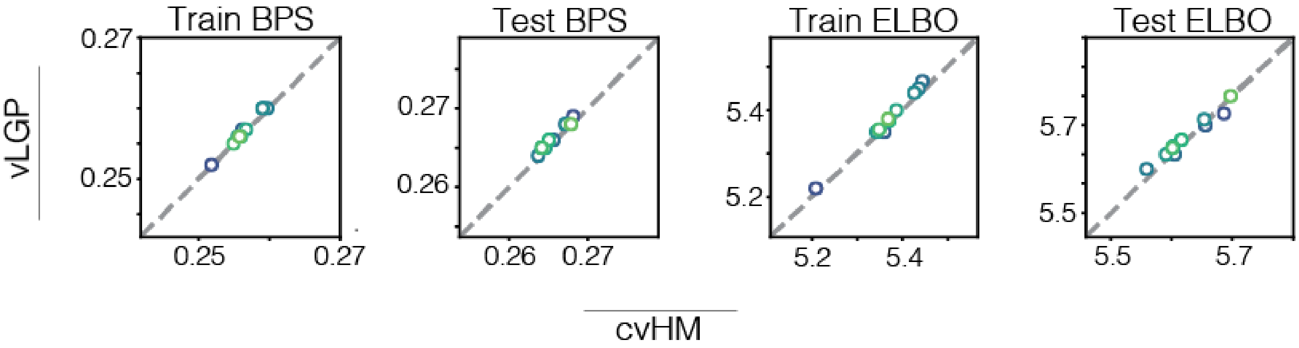

Figure 7: cvHM vs vLGP: BPS/normalized ELBO on train/test splits of the toy data used for verification. cvHM performs qualitatively the same as vLGP in terms of all metrics.

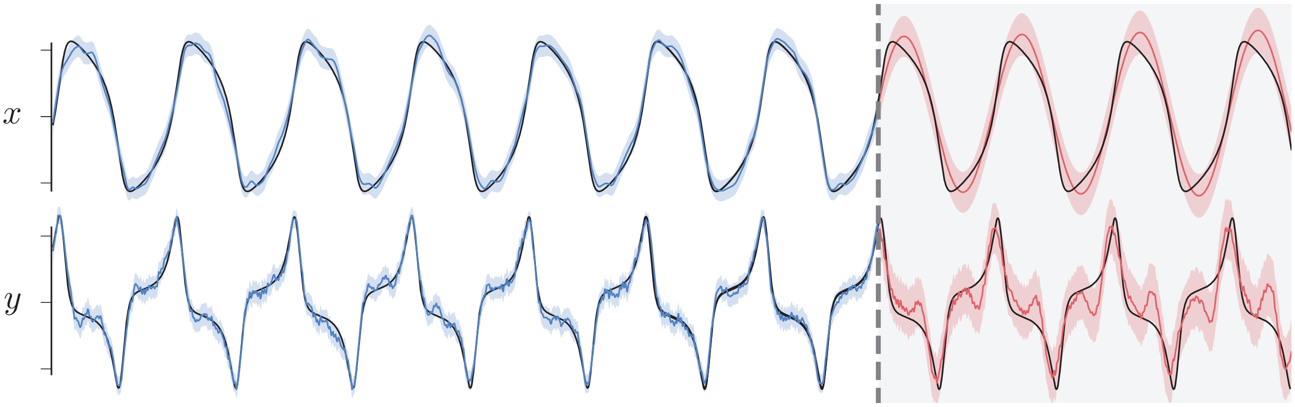

Figure 9: Van der Pol oscillator: cvHM predictions of future latent states of the. Before dash line: approximate posterior mean in blue, After dash line: predictive mean in red Colored shade: the 95% credible interval. Predictions of the y dimension are able to capture salient features such as the sharp peak and asymmetric behavior about it.

D.2 Van der Pol oscillator predictions

With the expressiveness of Hida-Matérn kernels, we can predict (forecast) future latent states. For the Van der Pol oscillator experiment we wondered how well cvHM could predict future latent states in the absence of data. Since the Van der Pol system evolves according to nonlinear dynamics, a sufficiently expressive covariance function is needed per dimension in order to make accurate predictions. We can easily construct expressive Hida-Matérn kernels through linear combination; we use 6, 2-ple Hida-Matérn kernels for the x dimension, and 30, 2-ple Hida-Matérn kernels for the y dimension.

The kernels over the y dimension are initialized so that they cover a range of frequencies from 0 Hz to 70 Hz. In order to isolate how expressive a kernel we can create, we further initialize the loading matrix and bias to the true values. During inference, we optimize all hyperparameters of all kernels. Fig. 9 shows that cvHM can perform prediction well at the cost of having a large expanded latent space.

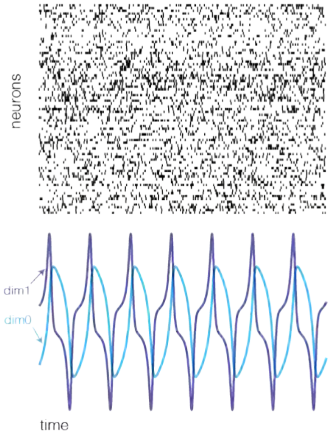

Figure 8: Observed spike counts and the 'true' latent trajectories for the VDP system.

D.3 DMFC-RSG

For the DMFC-RSG dataset, we analyzed all four conditions: 'hand-left,' 'hand-right,' 'eye-left,' 'eye-right' using cvHM, GPFA, and PLDS. Using the different methods we examine time instantaneous speed of neural trajectories either aligned to 'Set' or 'Go'. We fix the latent space to be three-dimensional for the purposes of visualization as well as qualitative metrics of performance. Fig. 10 shows that in terms of BPS and

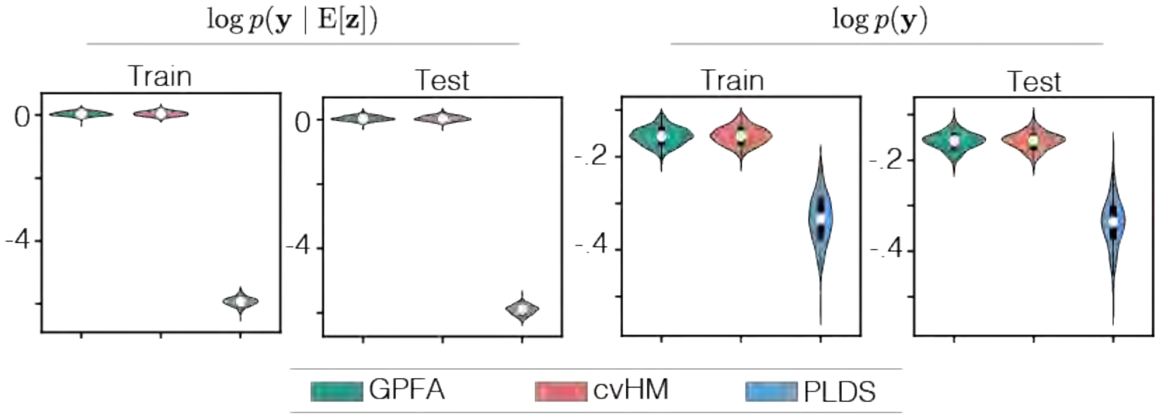

Figure 10: DMFC-RSG: BPS and log-marginal-likelihood for GPFA, cvHM, and PLDS per trial. Log-marginal-likelihood is calculated using a Monte-Carlo estimate with 100,000 samples from the posterior. A soft rectification was used in order to transform the output of GPFA to fitting rate for log-marginal-likelihood.

log-marginal-likelihood, cvHM and GPFA perform similarly while PLDS lags behind. Be aware that the true dynamics in nonlinear and cvHM and GPFA have a larger effective dimensionality in SSM point of view. In section K, the inferred speed of latent trajectories by all the three methods are plotted in Fig. 16, 17, 18,and 19.

D.4 MC-RTT

To see how ‘trial splitting’ affects the hyperparameter tuning, we examined the estimation of length scale using the MC-RTT dataset where we used a spike train of 20,000 bins but split it into trials of lengths 25, 50, 250, and 1250 bins. In Fig. 11, we see that the optimal length scale inferred for one of the latent dimensions decreases monotonically with increasing trial lengths.

E cvHM implementation details

E.1 Initialization

For cvHM, we initialize the readout matrix, $\mathbf{C}$, using factor analysis and the bias, $\mathbf{b}$, using the average firing rate. Except for the first experiment to compare performance with vLGP, hyperparameters are optimized in variational EM style. For optimization of the readout matrix, bias, and kernel hyperparameters we use PyTorch (Paszke et al., 2019) in combination with SciPy (Virtanen et al., 2020).

E.2 cvHM Algorithm

We describe the exact cvHM learning algorithm that uses dual information filtering along with CVI for inference in Alg. 2. There, $\mathbf{Q}_\infty$ is the stationary covariance of the dynamical system defined by $(\mathbf{A}, \mathbf{Q})$, and $\mathbf{h}_0$ and $\mathbf{J}_0$ are the prior natural parameters; which in our case will be $\mathbf{0}$ and $\mathbf{K}_{TT}$ respectively.

E.3 Hyperparameters using log-marginal likelihood

Discussed in the main text, the ELBO (represented below), for the models considered can be recast as

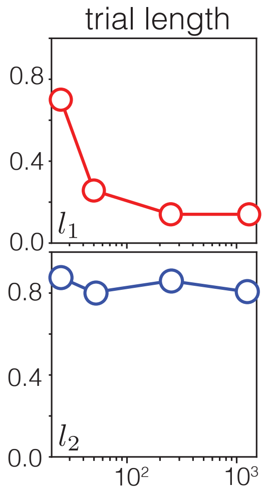

Figure 11: Inferred lengthscales of the two latent dimensions for MC-RTT as a function of trial length

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_t \mathbb{E}_{q(\mathbf{z}_t)} \log \frac{p(\mathbf{y}_t \mid \mathbf{z}_t)}{p(\tilde{\mathbf{y}}_t \mid \mathbf{z}_t)} + \sum_t \log p(\tilde{\mathbf{y}}_t \mid \tilde{\mathbf{y}}_{1:t-1}) \tag{64}$$

Algorithm 2 cvHM Algorithm

```
Input: $\mathbf{y}_{1:T}$, $\mathbf{V}_{1:T}$, $\mathbf{A}$, $\mathbf{Q}$, $\mathbf{A}^b$, $\mathbf{Q}^b$, $\mathbf{Q}_\infty$, $\mathbf{h}_0$, $\mathbf{J}_0$
initialization:
    *initialize the variational approximation to the prior*
    $\mathbf{h}_{1:T} \leftarrow \mathbf{h}_0$
    $\mathbf{J}_{1:T} \leftarrow \mathbf{J}_0$
    *initialize pseudo natural parameters*
    $\tilde{\mathbf{h}}_t \leftarrow \nabla_{\boldsymbol{\mu}_t^{(1)}} \mathbb{E}_{q(\mathbf{z}_t)}[\log p(\mathbf{y}_t \mid \mathbf{z}_t)], \quad t = 1, \ldots, T$
    $\tilde{\mathbf{J}}_t \leftarrow \nabla_{\boldsymbol{\mu}_t^{(2)}} \mathbb{E}_{q(\mathbf{z}_t)}[\log p(\mathbf{y}_t \mid \mathbf{z}_t)], \quad t = 1, \ldots, T$
repeat
    *bidirectional filtering*
    $(\mathbf{h}^f_{1:T}, \mathbf{J}^f_{1:T}) \leftarrow$ FilterForward$(\tilde{\mathbf{h}}_{1:T}, \tilde{\mathbf{J}}_{1:T}, \mathbf{A}, \mathbf{Q}, \mathbf{Q}_\infty)$
    $(\bar{\mathbf{h}}^b_{1:T}, \bar{\mathbf{J}}^b_{1:T}) \leftarrow$ FilterBackward$(\tilde{\mathbf{h}}_{1:T}, \tilde{\mathbf{J}}_{1:T}, \mathbf{A}^b, \mathbf{Q}^b, \mathbf{Q}_\infty)$
    $\mathbf{J}_t \leftarrow \mathbf{J}^f_{1:T} + \bar{\mathbf{J}}^b_{1:T} - \mathbf{Q}_\infty^{-1}$
    $\mathbf{h}_t \leftarrow \mathbf{h}^f_{1:T} + \bar{\mathbf{h}}^b_{1:T}$
    *update pseudo natural parameters*
    $\tilde{\mathbf{h}}_t \leftarrow (1-\alpha)\tilde{\mathbf{h}}_t + \alpha \nabla_{\boldsymbol{\mu}_t^{(1)}} \mathbb{E}_{q(\mathbf{z}_t)}[\log p(\mathbf{y}_t \mid \mathbf{z}_t)], \quad t = 1, \ldots, T$
    $\tilde{\mathbf{J}}_t \leftarrow (1-\alpha)\tilde{\mathbf{J}}_t + \alpha \nabla_{\boldsymbol{\mu}_t^{(2)}} \mathbb{E}_{q(\mathbf{z}_t)}[\log p(\mathbf{y}_t \mid \mathbf{z}_t)], \quad t = 1, \ldots, T$
until convergence
Return: $\mathbf{J}_{1:T}, \mathbf{h}_{1:T}$
```

We see that, with parameters of the variational approximation fixed, only the log-marginal-likelihood of pseudo observations contributes to the gradient with respect to hyperparameters. Then,

$$
\begin{aligned}
\nabla_{\boldsymbol{\theta}} \mathcal{L} &= \nabla_{\boldsymbol{\theta}} \sum \log p(\tilde{\mathbf{y}}_t \mid \tilde{\mathbf{y}}_{1:t-1}) && (65) \\
&= \nabla_{\boldsymbol{\theta}} \sum \log \left(\int p(\tilde{\mathbf{y}}_t \mid \mathbf{z}_t^S) p(\mathbf{z}_t^S \mid \tilde{\mathbf{y}}_{1:t-1}) \right) && (66) \\
&= \nabla_{\boldsymbol{\theta}} \sum \log \left(\int \mathcal{N}(\tilde{\mathbf{y}}_t \mid \mathbf{z}_t, \tilde{\mathbf{V}}_t) \mathcal{N}(\mathbf{z}_t^S \mid \mathbf{m}_t^-, [\mathbf{P}_t^-]^{-1}) d\mathbf{z}_t \right) && (67) \\
&= \nabla_{\boldsymbol{\theta}} \Big(-\tfrac{1}{2} (\tilde{\mathbf{y}}_t - \mathbf{H}\mathbf{m}_t^-)^\top \mathbf{R}_t^{-1} (\tilde{\mathbf{y}}_t - \mathbf{H}\mathbf{m}_t^-) - \tfrac{1}{2} \log |\mathbf{R}_t| \Big) && (68)
\end{aligned}
$$

where $\mathbf{R}_t = \mathbf{H}\mathbf{P}_t^- \mathbf{H}^\top + \tilde{\mathbf{V}}_t$, and $\mathbf{m}_t^- = \mathbf{A}_{\boldsymbol{\theta}} \mathbf{m}_{t-1}$, with $\mathbf{P}_t^- = \mathbf{A}_{\boldsymbol{\theta}} \mathbf{P}_{t-1} \mathbf{A}_{\boldsymbol{\theta}}^\top + \mathbf{Q}_{\boldsymbol{\theta}}$ are the predictive means and covariances at time t, computed using the filtering step statistics; they depend on the kernel hyperparameters through the transition matrix, $\mathbf{A}$, and the state noise $\mathbf{Q}$. In order to verify hyperparameter optimization, we draw spike trains whose intensity are modulated by a Matérn 3/2 GP while varying the kernel lengthscale. We use variational EM to estimate the hyperparameters of the prior as shown in Fig. 12.

E.4 Poisson likelihood gradients

For the update to the first natural parameter we have

$$
\begin{aligned}
\mathbf{h}_k = (1-\alpha)\mathbf{h}_{k-1} &+ \alpha \mathbf{h}_0 \\
&+ \alpha \nabla_{\mathbf{m}(\boldsymbol{\lambda})} \sum_t \mathbb{E}_{q(\mathbf{z}_t \mid \boldsymbol{\lambda})} \log p(\mathbf{y}_t \mid \mathbf{z}_t) \qquad (69)
\end{aligned}
$$

letting $l_{n,t} = \mathbb{E}_{q(\mathbf{z}_t \mid \boldsymbol{\lambda})} \log p(y_{n,t} \mid \mathbf{z}_t)$, we have through the chain rule and natural parameter properties (Hamelijnck et al., 2021) that

$$
\begin{aligned}
\nabla_{\mathbf{m}(\boldsymbol{\lambda})} l_{n,t} &= \nabla_{\mathbf{m}} l_{n,t} - 2 \nabla_{\mathbf{P}} l_{n,t} \mathbf{m}_t && (70) \\
&= \mathbf{C}_n \left(y_{n,t} - \Delta r_{n,t} \right) + \Delta r_{n,t} \mathbf{C}_n \mathbf{C}_n^\top \mathbf{m}_t && (71)
\end{aligned}
$$

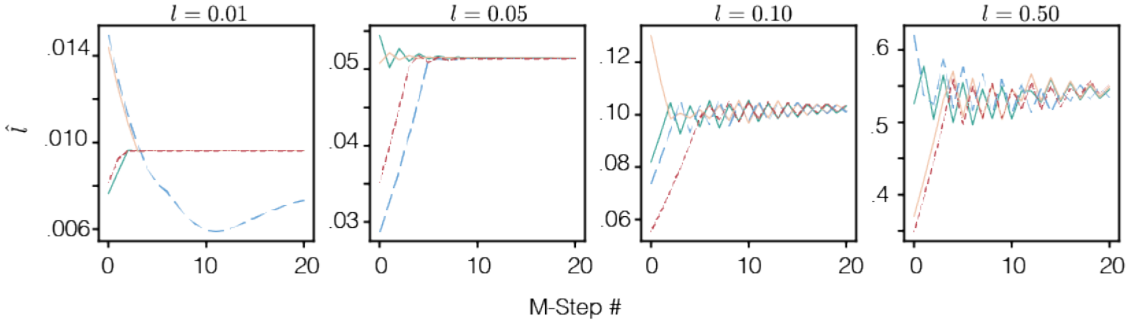

Figure 12: Optimal length scale estimated from the data during the M-steps. We initialized the prior length scale randomly (4 realizations) about the true value to see how optimization performed. In most cases, the hyperparameters converged with a relatively small number of M-steps. The oscillation is likely attributed to the fact that we cap the gradient within 10-25% of current value at each optimization step.

where $r_{n,t} = \exp(\mathbf{C}_n^\top + \mathbf{b}_n + \frac{1}{2}\mathbf{C}_n^\top \mathbf{P}_t \mathbf{C}_n)$. For the update to the second natural parameter we have

$$\begin{aligned} \mathbf{J}_k &= (1-\alpha)\mathbf{J}_{k-1} + \alpha \mathbf{J}_0 \\ &\quad + \alpha \nabla_{\boldsymbol{\Psi}(\boldsymbol{\lambda})} \sum_t \mathbb{E}_{q(\mathbf{z}_t \mid \boldsymbol{\lambda})} \log p(\mathbf{y}_t \mid \mathbf{z}_t) \end{aligned} \tag{72}$$

where

$$\nabla_{\boldsymbol{\Psi}(\boldsymbol{\lambda})} l_{n,t} = \nabla_{\mathbf{P}} \left(y_{n,t} \mathbf{C}_n^\top \mathbf{m}_t - \Delta r_{n,t} \right) \tag{73}$$

$$= -\tfrac{1}{2} \Delta \mathbf{C}_n \mathbf{C}_n^\top r_{n,t} \tag{74}$$

F Latent process velocity

A by-product of inference using the SSM representation of Hida-Matérn GPs is that all mean square derivatives of the latent trajectories are inferred *for free*. Those mean square derivatives are equal to the sample path derivatives with probability one (Doob, 1990) and may reveal useful information about latent trajectories inferred. One possibility, is using the mean square derivatives to probabilistically interpret the speed and acceleration at which neural trajectories evolve. In the context of neural dynamics this can be useful; as an example, the velocity of latent neural dynamics is an interesting metric that may be used to substantiate or disprove certain hypothesis of neural computation (Sohn et al., 2019). While similar conclusions could possibly be drawn through strategies like finite differences, they would not provide a calibrated measure of uncertainty. We examine this in the experiments section.

G PLDS with extended state-space

Our comparisons with PLDS in the paper used the same latent dimensionality. However, the state-space dimensionality of cvHM will necessarily be inflated due to the need to propagate mean square derivatives of the latent processes. In Fig. 13 we show the result of an additional experiment where PLDS is also given an extended state-space.

H Non-conjugate Gaussian observation example

To demonstrate the ability of cvHM to handle variety of non-conjugate cases, we consider nonlinear Gaussian observations readout from the latent state according to the following generative model

$$\mathbf{z}_{l,1:T} \sim \mathcal{GP}(0, k_l(\tau)) \qquad l = 1, \ldots, L \tag{75}$$

$$\mathbf{y}_{n,t} \sim \mathcal{N}(\mathbf{y}_{n,t} \mid g(\mathbf{z}_t), \sigma_n^2) \qquad n = 1, \ldots, N \tag{76}$$

$$g(\mathbf{z}_t) = \exp(\mathbf{C}_n^\top \mathbf{z}_t + \mathbf{b}_n) \tag{77}$$

Following the prescription earlier, the only adjustments we need to make to infer the latent trajectories are to calculate the derivatives of the expected log-likelihood under our variational approximation. Doing so, we have for

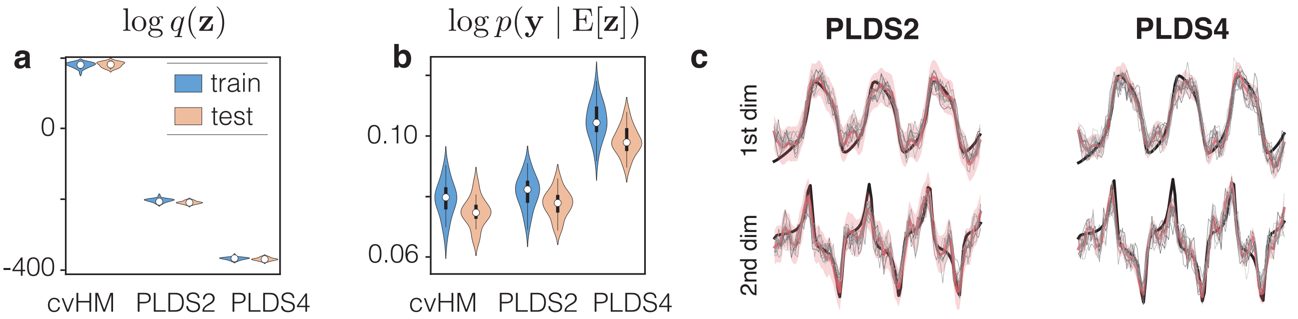

Figure 13: cvHM vs PLDS: For observations that read out L latent variables, the dimension of cvHM's latent space will be $M = \sum M_l$ when the l^{th} latent variable is modeled by an M_l-ple Hida-Matérn GP. We compare cvHM with $M_1 = 2$ and $M_2 = 2$ to PLDS with latent dimensionality of 2 and 4. (**a**) log-likelihood of the 'ground truth' under the posterior for cvHM and PLDS2 and PLDS4 (2 and 4 respectively denote the dimensionality of the latent space imposed). (**b**) bits-per-spike (**c**) comparison of posterior for PLDS2 and PLDS4; PLDS4 has been projected down from 4 dimensions to 2.

$l_{n,t} = \mathbb{E}_{q(\mathbf{z}_t|\boldsymbol{\lambda})} \log p(\mathbf{y}_{n,t} \mid \mathbf{z}_t)$, the gradients for the mean and covariance are:

$$\nabla_{\mathbf{m}_t} l_{n,t} = -\frac{1}{\sigma_n^2} \mathbf{C}_n r_{n,t} \left[r_{n,t} - \mathbf{y}_{n,t} \right]$$
$$\nabla_{\mathbf{P}_t} l_{n,t} = -\frac{1}{\sigma_n^2} \mathbf{C}_n \mathbf{C}_n^\top r_{n,t} \left[r_{n,t} - \mathbf{y}_{n,t} \right]$$

Fig. 14 shows the result of inference under this generative model when the true latent trajectories are generated according to the Lorenz system. In this case, we were able to analytically calculate the expected log-likelihoods; in general this may not always be possible, in which case we can resort to a sampling scheme to approximate the intractable expectations required for CVI.

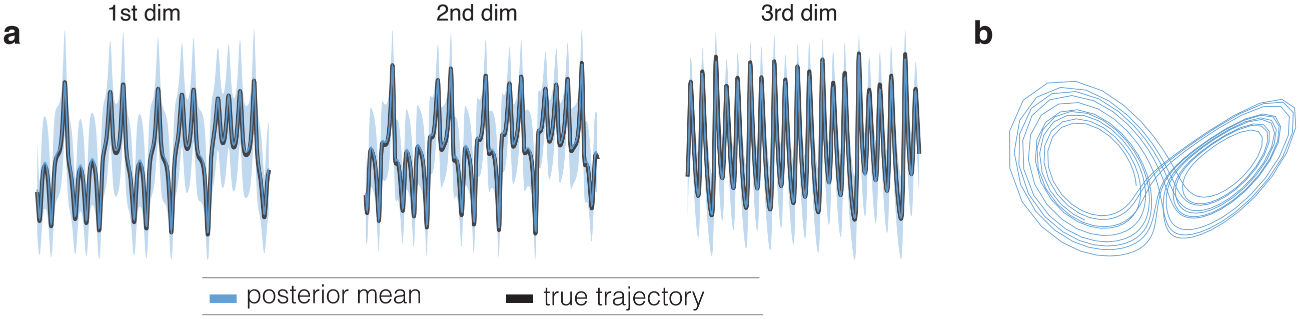

Figure 14: Nonlinear readout example: (**a**) cvHM infers accurate latents for the Lorenz system, observations are nonlinear readouts of a linear projection of the latent state; shading indicates the 95% credible interval (**b**) mean inferred trajectories plotted in 3D

I Comparisons against other Markovian GPs

In the main text, comparisons were made against models aimed at finding structured representations of data where prior beliefs are specified according to GPs (e.g. PLDS, GPFA, and vLGP). Here, we compare approximate GP regression with Hida-Matérn kernels versus popular Markovian GPs (Hartikainen & Sarkka, 2010; Solin et al., 2018; Hamelijnck et al., 2021); we choose two common baselines that use Poisson likelihoods from Wilkinson et al. (2020).

Coal dataset The coal mining dataset reports the 191 coal mining explosions that killed 10 or more people in Great Britain from 1851 to 1962 (Wilkinson et al., 2020). In order to make a fair comparison, we parameterize a GP similar to (Hamelijnck et al., 2021) by using a second order Hida-Matérn GP, and a Poisson likelihood with canonical link function so that the

	No. pts. (×1000)		
Likelihood	1	10	25
exact	-44.5 ± 14.4	-1.55 ± 0.72	-0.557 ± 0.680
whittle	**-8.5 ± 3.52**	**-1.02 ± 0.66**	**-0.408 ± 0.121**

Table 1: **Quantifying latent state recovery using Whittle versus exact log-likelihood**. In Fig. 3 we highlighted the difference between the loss landscape generated by the exact log-likelihood versus the Whittle approximation; here, we explore the Whittle approximation's effect on latent state recovery in the approximate inference setting. Numbers show the log-likelihood of the latent trajectories under the approximate posterior.

generative model is

$$p(\mathbf{z}_{1:T}) = \mathcal{N}(\mathbf{z}_{1:T} \mid \mathbf{0}, \mathbf{K}_{TT}) \tag{78}$$

$$p(y_t \mid z_t) = \text{Poisson}\left(y_t \mid \Delta \exp(z_t + b)\right) \tag{79}$$

where Δ is the bin size and b is the bias. We use the Whittle likelihood and bidirectional information filtering and calculate a 10 fold cross validated negative predictive log marginal likelihood of 0.955 ± 0.16, whereas the Markovian GP baselines used in (Wilkinson et al., 2020) achieve a 10 fold cross validated log marginal likelihood of 0.922 ± 0.11, when using the same data splits. cvHM performs worse on this dataset, possibly because the bias introduced by the Whittle approximation is significant in this low data regime. We show the inferred posterior intensity in Fig. 15**b**.

Aircraft accidents dataset. The aircraft accidents dataset is another dataset that is well modeled by a Poisson likelihood. For a fair comparison against the approximate GP regression methods reported in (Wilkinson et al., 2020), the GP prior kernel is constructed as

$$\begin{aligned} k(\tau) = \sigma_1^2 k_{H,M}(\tau; \tfrac{5}{2}, 0, \rho_1) + \sigma_2^2 k_{H,M}(\tau; \tfrac{3}{2}, b_2, \rho_2) \\ + \sigma_3^2 k_{H,M}(\tau; \tfrac{3}{2}, b_3, \rho_3) \end{aligned}$$

cvHM achieves a 10 fold cross validated negative predictive log marginal likelihood of 0.142 ± 0.01 on this dataset which is the same as results reported in (Wilkinson et al., 2020). As a consequence of this dataset being larger than the coal dataset, cvHM is able to achieve results on par with other Markovian GP baselines even though it uses an approximation of the Gaussian log-marginal likelihood. The inferred posterior intensity is plotted in Fig. 15**a**.

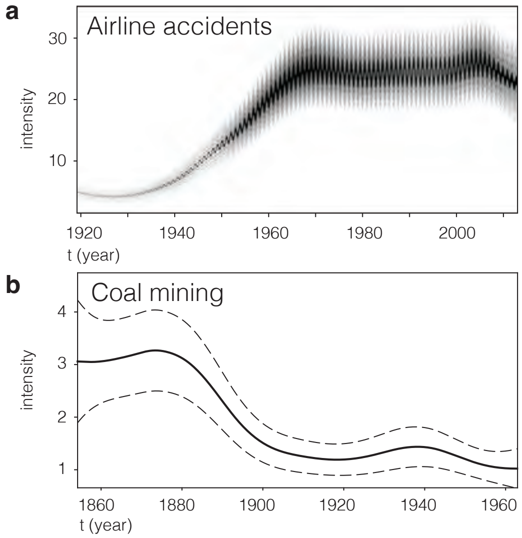

Figure 15: **a)** Mean airline accidents intensity and the 95% credible interval. **b)** Mean coal mining explosion accidents and 95% credible interval.

J Ablation studies

We tease apart some factors of cvHM that may make it preferable over a traditional/standard implementation that does not incorporate use of the Whittle likelihood or bidirectional information filtering with a few short experiments.

Whittle likelihood and latent variable recovery In the main paper, it was empirically shown that the Whittle approximation may produce a favorable loss landscape for hyperparameter optimization. Given that variational EM is an iterative algorithm, we would expect that improvements in hyperparameter estimates lead to improvements in the variational approximation to the posterior of latent trajectories.

Motivated by this, we consider a simple experiment where Poisson spiking is generated by a latent state simulated from a noisy Van der Pol oscillator like in Fig. 4. Over 3 random seeds, we generate datasets of different number of trials, then measure the average log probability of the inferred latent trajectories when using the Whittle approximation or exact log-marginal likelihood – the results are reported in Table 1.

Information filtering improves lower floating point results In this small experiment, we investigate how recovery of latent trajectories differs when using the information or covariance form of the Kalman filter. Data is generated from Poisson

Floating Pt.	Filter	Seed No.				
		1	2	3	4	5
32 bit	covariance	-52,035,944	-50,464,916	-52,000,688	-52,378,684	-51,895,720
	information	-52,030,328	-50,463,748	-51,997,296	-52,370,120	-51,893,184
64 bit	covariance	-51,920,061	-50,352,983	-51,884,765	-52,261,973	-51,780,153
	information	-51,920,061	-50,352,983	-51,884,765	-52,261,973	-51,780,153

Table 2: **Effect of filter type on latent variable recovery**. The average log-probability of latent trajectories for each seed is plotted as a function of the type of filter used, and the floating point precision. Only when dropping the floating point precision down to 32 bits do we see the benefit of using the information filter over the standard covariance filter.

observations with Van der Pol dynamics, similar to the previous experiment; over 5 random seeds, we draw 15 trials of length 1000, and then perform inference using either the information form of the Kalman filter, or the covariance form.

For each seed, the average log probability of latent trajectories is calculated and plotted in Table 2. For 64 bit floating point, there is no advantage to using the information filter. However, results for the covariance filter are consistently worse across each seed when the floating point precision is brought down to 32 bits.

K DMFC-RSG figures

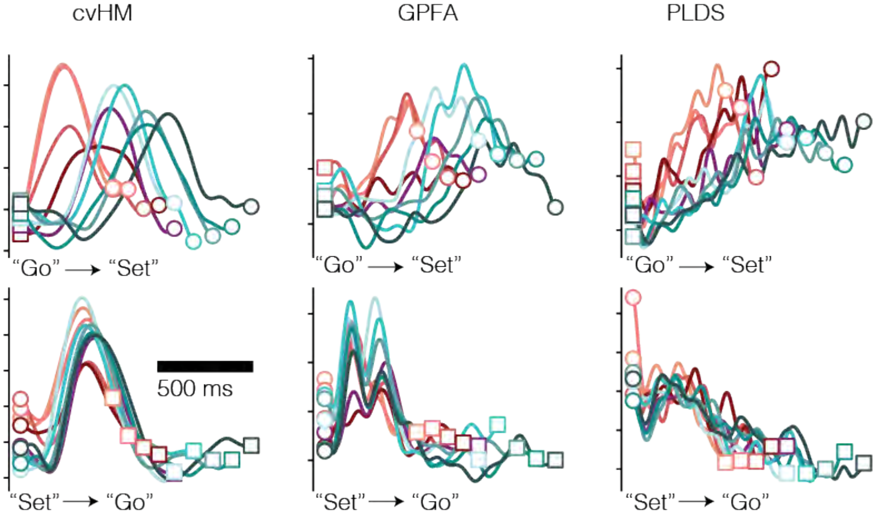

Figure 16: DMFC-RSG: **eye-left** condition. Similar to the eye-right condition presented in the main text, we can see that cvHM and GPFA recover latent trajectories which have peak speeds that decrease with respect to increasing intervals within the same prior. This effect is harder to see in the trajectories inferred by PLDS.

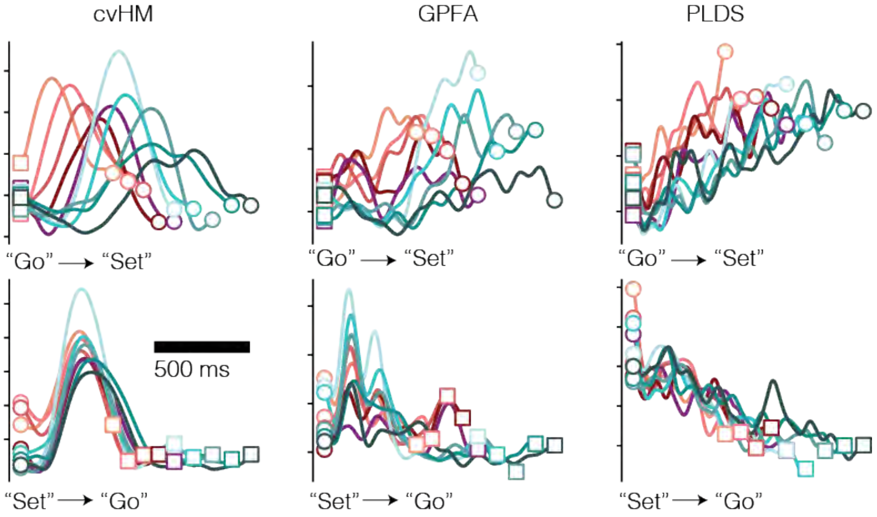

Figure 17: Condition: **eye-right**. Neural trajectory velocities for the eye-right condition as presented in the main paper. Again, PLDS seems to have inferred trajectories with an effective lengthscale that is too small.

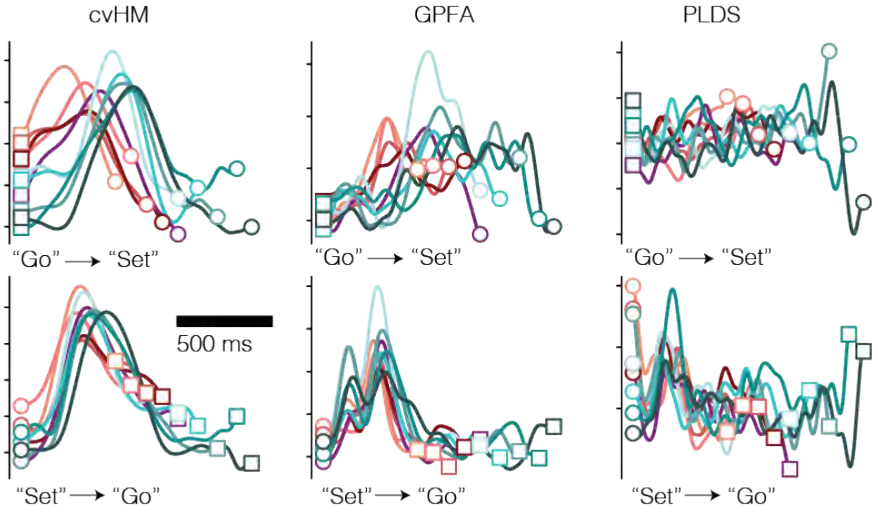

Figure 18: Condition: **hand-left**. In comparison to the conditions requiring an eye saccade to indicate interval predictions, trajectories seem to end at higher speeds as seen in 'Go' to 'Set'

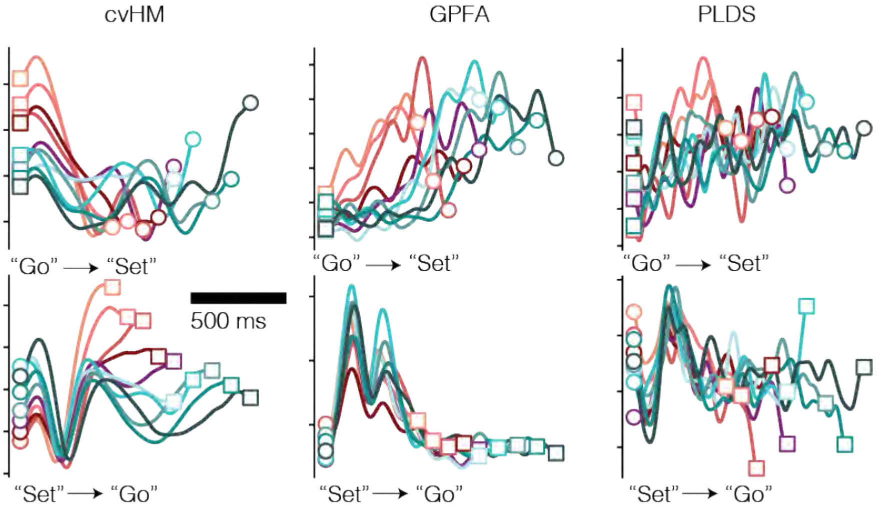

Figure 19: Condition: **hand-right**. cvHM inferred trajectories for this condition are distinctly different than those it inferred in the three other conditions; the 'bump' like characteristic of the speed exists in short prior but not so much the long prior; additionally the average trajectory speed dips from its starting value for all interval times.