Variational Gaussian Process Diffusion Processes

Prakhar Verma
Aalto University, Finland

Vincent Adam
University Pompeu Fabra, Spain

Arno Solin
Aalto University, Finland

Abstract

Diffusion processes are a class of stochastic differential equations (SDEs) providing a rich family of expressive models that arise naturally in dynamic modelling tasks. Probabilistic inference and learning under generative models with latent processes endowed with a non-linear diffusion process prior are intractable problems. We build upon work within variational inference, approximating the posterior process as a linear diffusion process, and point out pathologies in the approach. We propose an alternative parameterization of the Gaussian variational process using a site-based exponential family description. This allows us to trade a slow inference algorithm with fixed-point iterations for a fast algorithm for convex optimization akin to natural gradient descent, which also provides a better objective for learning model parameters.

1 INTRODUCTION

Continuous-time stochastic differential equations (SDEs, *e.g.*, Särkkä and Solin, 2019; Øksendal, 2003) are a ubiquitous modelling tool in fields ranging from physics (van Kampen, 1992) and finance (Eraker, 2001) to biology (Golightly and Wilkinson, 2011) and machine learning (Yildiz et al., 2018; Ruttor et al., 2013; García et al., 2017; Li et al., 2020b). SDEs offer a natural and flexible way to encode prior knowledge and capture the dynamic evolution of complex systems, where the stochasticity and nonlinearity of the underlying processes play a crucial role. In the particular setting where the drift of the SDE model is linear, the resulting prior process is a Gaussian process (GP, Rasmussen and Williams, 2006), known as a general and
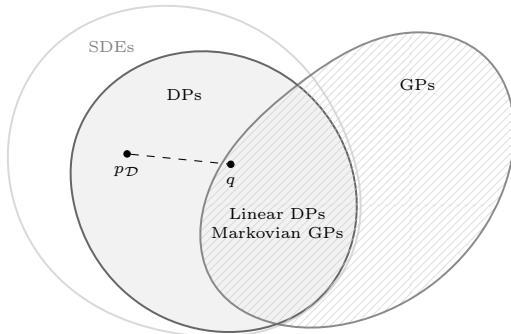
Figure 1: Approximating $p_\mathcal{D}$ with q for inference and learning in DPs.

powerful ML paradigm of its own. We focus on diffusion processes (DPs), which are a subset of SDEs with additional regularity conditions on the drift and diffusion functions (details in Higgs, 2011, Ch. 2). DPs with non-linear drifts cover a wide range of processes with *multi-modal*, *skew*, and *fat-tailed* behaviour (Fig. 2 left).

The generality of DPs, however, comes at a high practical cost: exact inference and parameter learning in non-linear DP models are computationally challenging or even intractable due to the infeasibility of direct simulation. Therefore, developing efficient and accurate methods for approximating DP models is crucial for theoretical and practical reasons. In this paper, we are concerned with Bayesian inference and learning in generative models with latent temporal processes endowed with an Itô DP prior.

Particularly, given a DP prior p and observations $\mathcal{D}$, we are interested in approximating the non-Gaussian DP posterior $p_\mathcal{D}$ with a linear-Gaussian DP q (Fig. 1). The seminal work by Archambeau et al. (2007a) used the framework of variational inference (VI, Blei et al., 2017) to derive an approximate inference algorithm, where the approximating variational family $\mathcal{Q}$ consists of time-variant linear (affine) DPs (*i.e.*, Markovian GPs, App. B in Rasmussen and Williams, 2006):

$$q : \mathrm{d}\mathbf{x}_t = f_q(\mathbf{x}_t, t)\,\mathrm{d}t + \mathbf{L}\,\mathrm{d}\boldsymbol{\beta}_t, \quad \mathbf{x}_0 \sim q(\mathbf{x}_0),$$
$$\text{s.t.} \quad f_q(\mathbf{x}_t, t) = \mathbf{A}_t\mathbf{x} + \mathbf{b}_t. \tag{1}$$

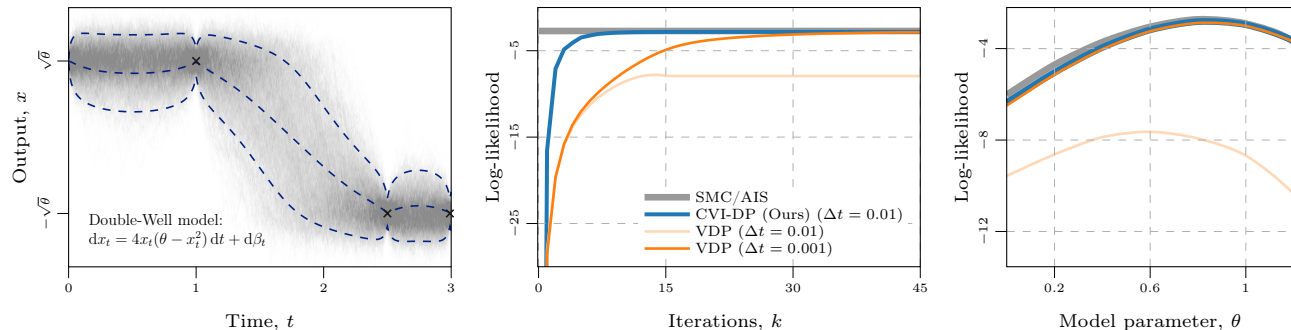They also introduced an objective for approximate

Figure 2: Left: Sequential Monte Carlo (SMC) samples and posterior of our CVI-DP for a non-linear diffusion process with skew and emerging modes between observations (✖). Middle: ELBO iterations highlight faster **inference** with CVI-DP vs. VDP. Right: Exact log-likelihood and ELBO as function of θ for parameter **learning**.

inference (the variational evidence lower bound or ELBO) and a fixed-point iteration algorithm to optimize it. In this paper, we highlight shortcomings in the method proposed by Archambeau et al. (2007a), which we refer to as VDP: *(i)* the inference algorithm is slow to converge even in the simple setting where the prior is a linear diffusion, and *(ii)* the parameterization of q via its drift function is ill-suited to the problem of learning the parameters of the prior diffusion from observations. To tackle these issues, we retain the problem formulation and objective but introduce an alternative site-based parameterization (Minka, 2001; Csató and Opper, 2002) and a new optimization algorithm. Crucially, the new parameterization allows us to trade the slow fixed-point algorithm of VDP for a fast and better-understood algorithm for convex optimization, akin to natural gradient descent (Amari, 1998). This stabilizes and speeds up inference (Fig. 2 middle) and aids parameter learning.

Contributions *(i)* We propose a novel site-based approach, the CVI-DP, for VI in models with latent DP priors that exploits the structure of the optimal Gaussian variational posterior; *(ii)* We show how CVI-DP speeds up inference and learning under both linear and non-linear DP priors; and *(iii)* We demonstrate the feasibility and efficiency of CVI-DP on a wide range of inference problems with DP priors featuring *multi-modal*, *skew*, and *fat-tailed* behaviours.

Related work For inference in general non-linear Gaussian sequential models, particle filtering a.k.a. sequential Monte Carlo (SMC, *e.g.*, Chopin et al., 2020) methods are popular tools. When computing the posterior given some observations (the *smoothing* problem), conditional particle filters (Andrieu et al., 2010; Whiteley, 2010; Lindsten et al., 2014) have proven reliable in avoiding mode-collapse and particle degeneracy (Svensson et al., 2015). For model parameter learning, automatic differentiation has enabled black-box learning of continuous-time dynamics (*e.g.*, Chen et al., 2018; Rubanova et al., 2019; Kidger et al., 2021). Close to our

work grounded in the framework of VI, Li et al. (2020b) introduced a tractable, sampling-based algorithm for inference and learning in general SDE models where the posterior process is parameterized via its non-linear drift and diffusion functions. Although general in scope, these methods are unnecessarily heavy and compute-intensive for many practical applications. Another related work is the VI algorithm by Wildner and Koeppl (2021) where the variational posterior is a controlled version of the prior (*i.e.*, the prior SDE with an extra term added to the drift). Although practical for inference, we argue our approach is better suited for learning.

We take an alternative route, trading some generality—by restricting the posterior process to a GP—for efficiency. We do so by explicitly 'Gaussianizing' the non-Gaussian observations and linearizing the non-linear diffusion. These principles are grounded in extended Kalman(–Bucy) filtering/smoothing (overview in Särkkä and Solin, 2019, Ch. 10) with numerous extensions (reviewed in Wilkinson et al., 2023) such as posterior linearization (García-Fernández et al., 2016, 2019).

This work aims at improving VDP originally proposed in Archambeau et al. (2007a,b) (overview in Sec. 2.1). The algorithm has been used as a building block, *e.g.* in drift estimation (Duncker et al., 2019), switching systems (Köhs et al., 2021), and learning neural network drift functions (Ryder et al., 2018). The work of Course and Nair (2023) introduces a Bayesian treatment of the DP learning problem by endowing the diffusion parameters with a prior distribution and deriving a mean-field variational algorithm to approximate the posterior over states and parameters. However, their formulation suffers from the same optimization issues as VDP, which is also reflected in the run time they report. Our approach turns the hard problem of VI under a DP prior into an easier problem of VI under a GP prior. For the latter, it is common to restrict the variational process to a GP (Challis and Barber, 2013). Efficient inference and learning that exploit the exponential family structure of the manifold of GPs and its geometry

can then be derived which are state of the art (Khan and Lin, 2017; Chang et al., 2020; Adam et al., 2021). These algorithms are equivalent to natural gradient descent (Amari, 1998) and exploit the structure of the optimal variational process in the natural parameterization (Sato, 2001), splitting the contribution of the prior and the observations into the posterior in an additive fashion. These previous approaches consider models with GP priors and non-Gaussian likelihoods, while we focus on the more general problem of DP priors.

2 BACKGROUND

An Itô diffusion process (DP) with state dimension d can be defined by an SDE as

$$p : \mathrm{d}\mathbf{x}_t = f_p(\mathbf{x}_t, t)\,\mathrm{d}t + \mathbf{L}\,\mathrm{d}\boldsymbol{\beta}_t, \quad \text{s.t.} \quad \mathbf{x}_0 \sim p(\mathbf{x}_0), \ (2)$$

where the drift $f_p : \mathbb{R}^d \times \mathbb{R}_+ \to \mathbb{R}^d$ is a non-linear mapping, the diffusion term $\mathbf{L}$ is linear (we drop dependency on t for notational convenience), and $\boldsymbol{\beta}_t$ denotes the Brownian motion with a spectral density $\mathbf{Q}_c$. The data $\mathcal{D} = \{(t_i, y_i)\}_{i=1}^n$ comprises input–output pairs and is assumed to constitute independent and identically distributed noisy versions of state trajectory $\mathbf{x}$ at n ordered discrete-time points via an observation model providing the likelihoods $\{p(y_i \mid \mathbf{x}_{t_i})\}_{i=1}^n$. The posterior process $p_{\mathcal{D}}$ can be shown to have the following structural properties: *(i)* it shares the same diffusion coefficient as the prior; *(ii)* it can be expressed as the sum of the prior drift f_p, and a data and prior dependent term g resulting from a backward pass through the process and observations (see App. A):

$$\begin{aligned} p_{\mathcal{D}} \ : \ &\mathrm{d}\mathbf{x}_t = (f_p(\mathbf{x}_t, t) + g(\mathbf{x}_t, t, \mathcal{D}, p))\,\mathrm{d}t + \mathbf{L}\,\mathrm{d}\boldsymbol{\beta}_t, \\ &\text{s.t.} \quad \mathbf{x}_0 \sim p_{\mathcal{D}}(\mathbf{x}_0). \end{aligned} \quad (3)$$

However, for most of the settings of interest, the posterior $p_{\mathcal{D}}$ is intractable and therefore approximate inference methods are used for both inference and learning.

VI turns the posterior inference problem into the maximization of a lower bound $\mathcal{L}(q)$ (ELBO) to the log marginal likelihood $\log p(\mathbf{y})$ with respect to a distribution q over the latent variable $\mathbf{x}$,

$$\log p(\mathbf{y}) \geq \mathbb{E}_q[\log p(\mathbf{y} \mid \mathbf{x})] - \mathrm{D}_{\mathrm{KL}}[q \,\|\, p] = \mathcal{L}(q), \quad (4)$$

where p is the prior distribution over $\mathbf{x}$, $p(\mathbf{y} \mid \mathbf{x})$ is the likelihood, and $\mathrm{D}_{\mathrm{KL}}[q \,\|\, p]$ is the Kullback–Leibler divergence between q and p. The gap in the inequality can be shown to be $\log p(\mathbf{y}) - \mathcal{L}(q) = \mathrm{D}_{\mathrm{KL}}[q \,\|\, p_{\mathcal{D}}]$. Thus, the bound is tight for $q = p_{\mathcal{D}}$, *i.e.* when q is the posterior distribution. We use probability density functions to refer to the associated distributions, as is commonly done in the field.

In the case of diffusion processes, the KL divergence between the variational process $q(\mathbf{x})$ and the prior process $p(\mathbf{x})$ can be expressed by using Girsanov theorem (Girsanov, 1960) leading to the ELBO,

$$\begin{aligned} \mathcal{L}(q) = \ &\mathbb{E}_{q(\mathbf{x})} \log p(\mathbf{y} \mid \mathbf{x}) - \mathrm{D}_{\mathrm{KL}}[q(\mathbf{x}_0) \,\|\, p(\mathbf{x}_0)] \\ &- \tfrac{1}{2} \int_0^T \mathbb{E}_{q(\mathbf{x}_t)} \| f_q(\mathbf{x}_t, t) - f_p(\mathbf{x}_t, t) \|_{\mathbf{Q}_c^{-1}}^2 \mathrm{d}t, \quad (5) \end{aligned}$$

where $\| \cdot \|_{\mathbf{Q}_c^{-1}}^2$ is the weighted 2-norm associated to the inner product $\langle \mathbf{f}, \mathbf{g} \rangle_{\mathbf{Q}_c^{-1}} = \mathbf{f}^\top \mathbf{Q}_c^{-1} \mathbf{g}$, and we set the diffusion function of the posterior process to its optimal value, which is that of the prior.

The prior DP might have free *model* parameters $\boldsymbol{\theta}$ (*e.g.* parameterizing the drift function as in Fig. 2) that need to be adjusted to best explain the observations, a task we refer to as the learning problem. In this scenario, the ELBO $\mathcal{L}(q, \boldsymbol{\theta})$ depends on both the model parameters $\boldsymbol{\theta}$ and the variational distribution q. Noting $q^*(\boldsymbol{\theta}) = \arg\max_q \mathcal{L}(q, \boldsymbol{\theta})$, the optimal variational distribution for fixed model parameters $\boldsymbol{\theta}$, a common objective to solve the learning problem is to maximize objective $\mathcal{L}(q^*(\boldsymbol{\theta}); \boldsymbol{\theta})$. This nested optimization problem is intractable and usually replaced by coordinate ascent of the ELBO with respect to $(q, \boldsymbol{\theta})$, a procedure known as variational Expectation–Maximization (VEM, Neal and Hinton, 1998, see App. E.2). The efficacy of VEM strongly depends on the choice of parameterization for the variational distribution q and its dependence on the parameters $\boldsymbol{\theta}$ (Adam et al., 2021).

2.1 Gaussian VI for Diffusion Process (VDP)

Archambeau et al. (2007a) propose to restrict the variational process $q(\mathbf{x})$ to be a diffusion with affine drift

$$\mathcal{Q} = \{q : \mathrm{d}\mathbf{x}_t = (\mathbf{A}_t \mathbf{x}_t + \mathbf{b}_t)\,\mathrm{d}t + \mathbf{L}\,\mathrm{d}\boldsymbol{\beta}_t, \ \mathbf{x}_0 \sim q(\mathbf{x}_0)\}, \ (6)$$

with $\mathbf{A}_t \in \mathbb{R}^{d \times d}$ and $\mathbf{b}_t \in \mathbb{R}^d$, corresponding to the set of Markovian Gaussian processes (*cf.* Särkkä and Solin, 2019, Ch. 12). In this setting, the marginal distribution $q(\mathbf{x}_t)$ of the process are fully characterised by the mean and covariance, $\mathbf{M} = (\mathbf{m}_t, \mathbf{S}_t)$.

Constrained optimization Archambeau et al. express the ELBO Eq. (5) in terms of both the variational parameters $\mathbf{V} = (\mathbf{A}_t, \mathbf{b}_t)$ specifying the drift function f_q and the marginal statistics $\mathbf{M}$ of the variational process q, which are coupled through ODEs:

$$C[\mathbf{M}, \mathbf{V}](t) = \begin{bmatrix} \dot{\mathbf{m}}_t - \mathbf{A}_t\,\mathbf{m}_t - \mathbf{b}_t \\ \dot{\mathbf{S}}_t - \mathbf{A}_t\,\mathbf{S}_t - \mathbf{S}_t\,\mathbf{A}_t^\top - \mathbf{Q}_c \end{bmatrix} = \mathbf{0} \quad \forall t. \ (7)$$

They introduce an alternative formulation of the ELBO optimization problem as the maximization of

$$\begin{aligned} \mathcal{L}(\mathbf{M}, \mathbf{V}) = \ &\mathbb{E}_{q(\mathbf{x})} \log p(\mathbf{y} \mid \mathbf{x}) - \mathrm{D}_{\mathrm{KL}}[q(\mathbf{x}_0) \,\|\, p(\mathbf{x}_0)] \\ &- \tfrac{1}{2} \int_0^T \mathbb{E}_{q(\mathbf{x}_t)} \| (\mathbf{A}_t \mathbf{x}_t + \mathbf{b}_t) - f_p(\mathbf{x}_t, t) \|_{\mathbf{Q}_c^{-1}}^2 \mathrm{d}t, \quad (8) \end{aligned}$$

subject to constraint $C[\mathbf{M}, \mathbf{V}](t) = \mathbf{0}, \forall t$. They propose to solve this constrained optimization problem via the method of Lagrangian multipliers (see App. C). This approach does not lead to a closed-form expression for the solution, but gives stationarity conditions for the optimal variational parameters,

$$(\mathbf{A}_t^*, \mathbf{b}_t^*) = \Pi_{q^*(\mathbf{x}_t)}[f_p(\cdot, t)] + g(\mathcal{D}, \mathbf{A}^*, \mathbf{b}^*, t), \quad (9)$$

where $\Pi_{q(\mathbf{x}_t)}[f_p(\cdot, t)]$ is the posterior linearization operator applied to the prior drift $f_p(\cdot, t)$ defined by

$$\Pi_{q(\mathbf{x}_t)}[f_p(\cdot, t)] = \arg\min_{(\mathbf{A}_t, \mathbf{b}_t)} \mathbb{E}_{q(\mathbf{x}_t)}\left[\|\mathbf{A}_t\mathbf{x}_t + \mathbf{b}_t - f_p(\mathbf{x}_t, t)\|_{\mathbf{Q}_c^{-1}}^2\right]. \quad (10)$$

Informally, the operator finds the best linear approximation of the drift in a squared loss sense, in expectation over the ongoing variational process.

Fixed point iterative algorithm These stationarity conditions imply an iterative algorithm,

$$(\mathbf{A}_t^{(k+1)}, \mathbf{b}_t^{(k+1)}) = \Pi_{q^{(k)}(\mathbf{x}_t)}[f_p(\cdot, t)] + g(\mathcal{D}, \mathbf{A}_t^{(k)}, \mathbf{b}_t^{(k)}, t), \quad (11)$$

for finding the variational parameters. An appealing property of these updates is that they mimic (up to the posterior linearization) the additive expression of the exact posterior drift in Eq. (3) (see App. C). As a follow-up, Archambeau and Opper (2011) clarified the connection to posterior linearization, which is also used as an explicit sub-routine in various approximate inference algorithms (García-Fernández et al., 2015; Tronarp et al., 2018).

Limitations of the VDP method The first issue is with the iterative algorithm in Eq. (11). The iterative algorithm is introduced out of convenience because it leads to closed-form updates. However, it does not come with convergence guarantees. Another issue stemming from the stationarity conditions Eq. (9) is that unlike in the exact inference case Eq. (3), the deviation g from the prior drift depends on the optimal posterior $(\mathbf{A}^*, \mathbf{b}^*)$ instead of on the prior p. This dependence makes the fixed-point updates arbitrarily slow to converge—even for simple linear diffusion (see Fig. A8).

Consequently, learning the model parameters (via VEM) becomes problematic. Following Adam et al. (2021), we argue that when learning parameters via ELBO maximization, the best parameterization of the variational posterior q is one that completely decouples the contribution of the prior from that of the observations. Such a decoupling occurs in conjugate posterior inference in exponential family models: the posterior natural parameters sum contributions of the prior and of the observations. This property of exponential family models is the building block of the method we propose. It is not possible to achieve

such a decoupling when parameterizing q via its drift function. This is true even when exploiting the additive expression of the exact posterior drift Eq. (3) because the deviation term g still mixes the prior and the observations. The algorithm proposed in Archambeau et al. (2007a) inherits this more general problem of parameterization via the drift function.

In the next section, we describe our method and how it fixes the aforementioned issues by exploiting the exponential family structure of linear diffusion processes. By parameterizing the posterior in terms of its natural parameters, we *(i)* bypass the need to use a fixed point algorithm, trading it for a well-understood convex optimization algorithm, and *(ii)* achieve a better separation of the prior and observation contributions to the posterior, speeding up the dynamics of learning.

In short, our approach hinges on the following steps: *(i)* We frame our method within the framework of VI, restricting the variational process to the set of linear diffusions; *(ii)* such a variational process belongs to an exponential family, and we parameterize it via its natural parameters; *(iii)* we speed up inference via natural gradient descent and ease learning by incorporating iterative posterior linearization of the prior in the framework.

3 CVI FOR LINEAR DPs (CVI-DP)

Consider the following continuous-discrete Gaussian diffusion model with Gaussian observations:

$$d\mathbf{x}_t = (\mathbf{A}_t\mathbf{x}_t + \mathbf{b}_t)\, dt + \mathbf{L}\, d\boldsymbol{\beta}_t, \quad y_i \,|\, \mathbf{x} = \mathbf{h}^\top \mathbf{x}_{t_i} + \epsilon_i, \quad (12)$$

with $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$ and $\mathbf{h} \in \mathbb{R}^d$ (example of such a model in Fig. A8). The diffusion process can be marginalized to state evaluations $\{\mathbf{x}_i = \mathbf{x}_{t_i}\}_{i=1}^n$ leading to the discrete-time Markov chain

$$\mathbf{x}_{i+1} = \hat{\mathbf{A}}_i\mathbf{x}_i + \hat{\mathbf{b}}_i + \hat{\epsilon}_i, \quad \hat{\epsilon}_i \sim \mathrm{N}(\mathbf{0}, \hat{\mathbf{Q}}_i), \quad (13)$$

where $(\hat{\mathbf{A}}_i, \hat{\mathbf{b}}_i, \hat{\mathbf{Q}}_i)$ are available in closed-form.

Exact inference for Gaussian observations The marginalized prior process belongs to the exponential family $\mathcal{F}$ which we define as containing non-degenerate Gaussians with probability density

$$p(\mathbf{x}_0, \ldots, \mathbf{x}_n) = \exp[\langle \mathsf{T}(\mathbf{x}), \boldsymbol{\eta}_p \rangle - A(\boldsymbol{\eta}_p)], \quad (14)$$

where $\mathsf{T}(\mathbf{x}) = [\mathbf{x}, \mathrm{btd}(\mathbf{x}\mathbf{x}^\top)]$ is the set of sufficient statistics with $\mathrm{btd}(\mathbf{M})$ sets entries of $\mathbf{M}$ outside of the d-block tri-diagonals to zero (see App. D), and $\boldsymbol{\eta}_p$ are the natural parameters. The likelihood factors $\{p(y_i \,|\, \mathbf{x}_i)\}_{i=1}^n$ can be expressed as proportional to the density of an exponential family distribution conjugate to $\mathcal{F}$ with sufficient statistics $\{\mathsf{T}(\mathbf{h}^\top \mathbf{x}_i)\}_{i=1}^n$ and natural parameters $\boldsymbol{\lambda}^* = (\sigma^{-2}\mathbf{y}, -\frac{1}{2}\mathbf{1}_n\sigma^{-2}) \in \mathbb{R}^{n \times 2}$. Due to the conjugacy, the posterior distribution $p(\mathbf{x} \,|\, \mathbf{y}) \propto p(\mathbf{y} \,|\, \mathbf{x})\, p(\mathbf{x})$
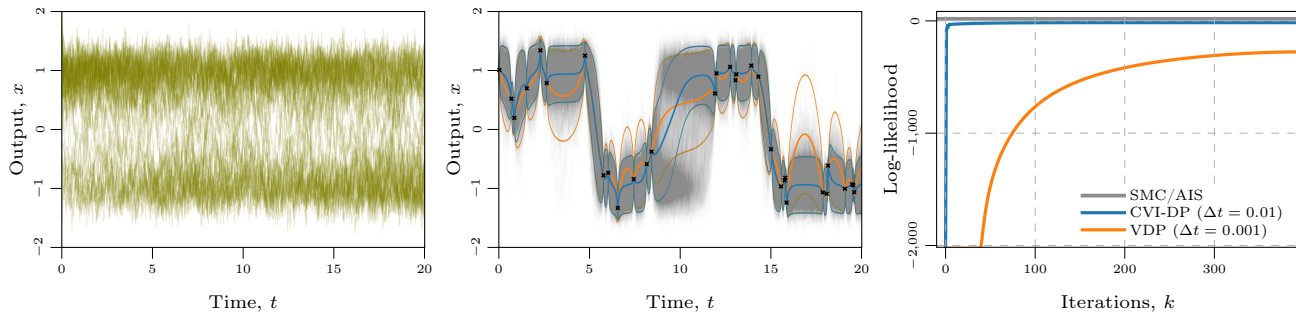
Figure 3: Approximate inference under a Double-Well prior (draws from prior on the left). Middle: Approximate posterior processes for CVI-DP and VDP overlaid on the SMC ground-truth samples. Right: Our CVI-DP converges quickly even with large discretization step when inferring the variational parameters, while VDP suffers from slow convergence even with a small discretization step.

belongs to $\mathcal{F}$. Its natural parameters are available in closed form and decompose additively as $\boldsymbol{\eta}_{\mathcal{D}} = \boldsymbol{\eta}_p + \phi(\boldsymbol{\lambda}^*)$, where $\phi(\boldsymbol{\lambda})$ is the linear mapping such that $\sum_i \langle \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i), \boldsymbol{\lambda}_i \rangle = \langle \mathsf{T}(\mathbf{x}), \phi(\boldsymbol{\lambda}) \rangle$. Importantly, $\boldsymbol{\lambda}^*$ is independent of the prior parameters $\boldsymbol{\theta}$.

Non-conjugate case If observations are independent conditioned on the process but no longer Gaussian distributed, one can resort to VI and show that the optimal Gaussian variational posterior $q \in \mathcal{F}$ has the same additive natural parameter structure $\boldsymbol{\eta}_{q^*} = \boldsymbol{\eta}_p + \phi(\boldsymbol{\lambda}^*)$. To optimize the ELBO, Mirror descent in the *expectation* parameterization of the variational distribution $\boldsymbol{\mu} = \mathbb{E}_q[\mathsf{T}(\mathbf{x})]$, with the Kullback–Leibler divergence as Bregman divergence (see App. E.1) provides an efficient algorithm to find the optimal $\boldsymbol{\lambda}^*$, via the local updates

$$\boldsymbol{\lambda}_i^{(k+1)} = (1-\rho)\,\boldsymbol{\lambda}_i^{(k)} + \rho\,\phi_i^{-1}\big(\nabla_{\boldsymbol{\mu}_i}\mathbb{E}_{q^{(k)}}\log p(y_i\,|\,\mathbf{x}_i)\big),\ (15)$$

where ϕ_i are the linear mappings such that $\langle \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i), \boldsymbol{\lambda}_i \rangle = \langle \mathsf{T}(\mathbf{x}_i), \phi_i(\boldsymbol{\lambda}_i) \rangle$. This procedure corresponds to the CVI inference algorithm introduced in Khan and Lin (2017).

Learning To learn the hyperparameters $\boldsymbol{\theta}$, we follow Adam et al. (2021) who proposed a modified VEM algorithm running coordinate ascent on $l(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \mathcal{L}(\boldsymbol{\eta}_p(\boldsymbol{\theta}) + \phi(\boldsymbol{\lambda}), \boldsymbol{\theta})$, instead of the usual coordinate ascent directly on $\mathcal{L}(\boldsymbol{\eta}_q, \boldsymbol{\theta})$, and showed that it speeds up the learning (see App. E.2). Informally, with the former loss, the optimal variational parameters $\boldsymbol{\lambda}^*(\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\lambda}} l(\boldsymbol{\lambda}, \boldsymbol{\theta})$ obtained for a fixed parameter setting $\boldsymbol{\theta}$ are more independent of $\boldsymbol{\theta}$ than in alternative parameterizations. As a consequence, they generalize better to the new parameters settings encountered during the learning, in the sense that they remain closer to the new optimal variational parameters across the M-steps iterations: $\boldsymbol{\lambda}^*(\boldsymbol{\theta}_{t+1}) \approx \boldsymbol{\lambda}^*(\boldsymbol{\theta}_t)$. Thus fewer adjustments of $\boldsymbol{\lambda}$ are needed in the E-steps during the learning process via VEM, which explains the faster convergence. As an extreme example, consider running inference and learning via coordinate

ascent $l(\boldsymbol{\lambda}, \boldsymbol{\theta})$ in the model with Gaussian observations described above. Since $\boldsymbol{\lambda}^*$ is independent of $\boldsymbol{\theta}$, it is only required to run Eq. (15) to convergence once to obtain $\boldsymbol{\lambda}^*$, and then to optimize the loss with respect to $\boldsymbol{\theta}$. Note that in this setting, it is unnecessary to resort to approximate inference since the marginal likelihood is available in closed form.

4 CVI-DP FOR NON-LINEAR DPs

Similar to the algorithm proposed in Archambeau et al. (2007a), we adopt the variational approach and restrict the variational process q to the set of linear diffusions. However, to derive our algorithm, we take a different route than Archambeau et al. (2007a). We first discretize the continuous-time inference problem effectively turning the continuous-time problem into a discrete-time problem. We then derive an inference algorithm for the discretized problem. Finally, we take a continuous-time limit of the resulting algorithm. This construction is similar to the work of Cseke et al. (2013).

4.1 Discretizing the Inference Problem

We discretize the prior process on an ordered time-grid $\tau = (t_1, \ldots, t_M) \in [0, T]$ using Euler–Maruyama as

$$p(\{\mathbf{x}\}_\tau) = p(\mathbf{x}_0) \prod_{m=0}^{M-1} \mathrm{N}(\mathbf{x}_{m+1}; \mathbf{x}_m + f_p(\mathbf{x}_m, m)\Delta t, \mathbf{Q}_c \Delta t),$$
(16)

where we denote the discretized process as $\{\mathbf{x}\}_\tau$. We assume the discretized process is observed at $n < M$ time points and denote by $\mathbf{x}^{\mathrm{d}}$ the vector of states such that $p(y_i\,|\,\{\mathbf{x}\}_\tau) = p(y_i\,|\,\mathbf{x}_i^{\mathrm{d}})$. The CVI algorithm applies to inference problems with exponential family priors. We denote by $\mathcal{F}_c$ the set of Markovian Gaussian processes marginalized to $\{\mathbf{x}\}_\tau$, which is a Gaussian exponential family with sufficient statistics
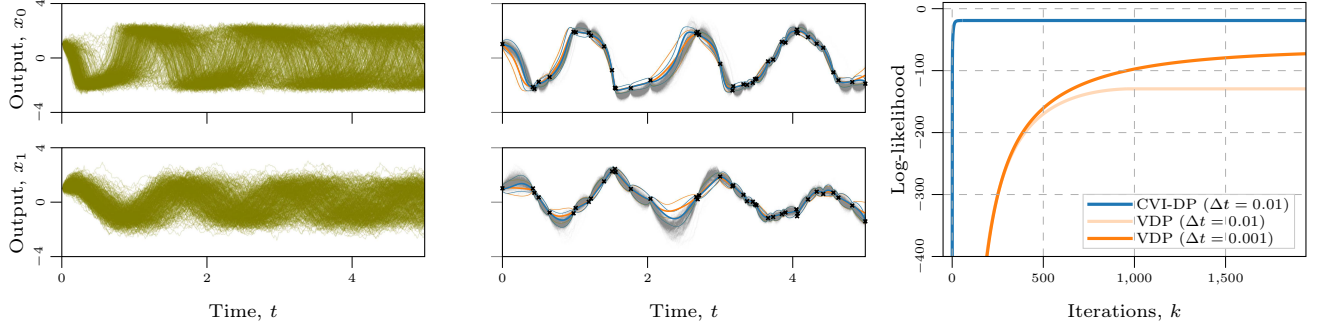
Figure 4: Approximate inference under a stochastic van der Pol oscillator prior (draws from prior on the left). Middle: Approximate posterior processes for CVI-DP and VDP overlaid on the SMC ground-truth samples. Right: CVI-DP converges quickly even with large discretization step when inferring the variational parameters, while VDP suffers from slow convergence even with a small discretization step. SMC did not converge with the provided budget and requires more number of particles in the multi-dimensional setup.

$\mathsf{T}(\{\mathbf{x}\}_\tau)$. We introduce $p_L \in \mathcal{F}_c$ with linear drift $f_L(\mathbf{x}_m, m) = \mathbf{A}_m \mathbf{x}_m + \mathbf{b}_m$ and the added constraint that it shares the same innovation noise as p as well as the same marginal distribution over the initial state $\mathbf{x}_0$. This allows to introduce the change of measure

$$\frac{p(\{\mathbf{x}\}_\tau)}{p_L(\{\mathbf{x}\}_\tau)} = \exp\Big(\sum_{m=0}^{M-1} \log \frac{p(\mathbf{x}_{m+1} \mid \mathbf{x}_m)}{p_L(\mathbf{x}_{m+1} \mid \mathbf{x}_m)}\Big)$$
$$= \exp\Big(\sum_{m=0}^{M-1} V(\tilde{\mathbf{x}}_m)\,\Delta t\Big), \qquad (17)$$

where $\tilde{\mathbf{x}}_m = [\mathbf{x}_{m+1}, \mathbf{x}_m]$ denotes consecutive pairs of states, and the potentials $V(\tilde{\mathbf{x}}_m)$ are given by

$$V(\tilde{\mathbf{x}}_m) = \frac{1}{2}\|f_L(\mathbf{x}_m) - f_p(\mathbf{x}_m)\|_{\mathbf{Q}_c^{-1}}^2 +$$
$$\Delta t \Big\langle \frac{\mathbf{x}_{m+1} - \mathbf{x}_m}{\Delta t} - f_L(\mathbf{x}_m), f_p(\mathbf{x}_m) - f_L(\mathbf{x}_m) \Big\rangle_{\mathbf{Q}_c^{-1}}. \quad (18)$$

Note that we dropped the time dependence of the drift function to simplify the notation. Details of the derivation are provided in App. F.1. We obtain an alternative expression for the posterior distribution

$$p(\{\mathbf{x}\}_\tau \mid \mathcal{D}) = p_L(\{\mathbf{x}\}_\tau)\, p(\mathbf{y} \mid \{\mathbf{x}\}_\tau)$$
$$\exp\Big(\sum_{m=0}^{M-1} V(\tilde{\mathbf{x}}_m)\,\Delta t\Big). \quad (19)$$

This corresponds to an inference problem in which the prior over the latent chain has the simpler density p_L, and two likelihoods: the original and now *sparse* likelihood $p(\mathbf{y} \mid \{\mathbf{x}\}_\tau)$ and an additional *dense* likelihood stemming from the change of measure. We are now back in the setting of models with linear diffusions as priors, where VI can be achieved via the Mirror descent algorithm introduced in Sec. 3.

4.2 Solving the Discretized Problem

In the discrete form, the intractable posterior can be approximated by a Markovian Gaussian process parameterized via a mixture of dense and sparse sites as done

in Cseke et al. (2013, 2016):

$$q(\{\mathbf{x}\}_\tau) \propto p_L(\{\mathbf{x}\}_\tau)\exp\Big(\sum_{i=1}^{n}\langle \boldsymbol{\lambda}_i, \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i^{\mathrm{d}})\rangle$$
$$+ \sum_{m=0}^{M-1}\langle \boldsymbol{\Lambda}_m, \mathsf{T}(\tilde{\mathbf{x}}_m)\rangle \Delta t\Big). \quad (20)$$

The approximating distribution q belongs to $\mathcal{F}_c$ and has natural parameters $\boldsymbol{\eta}_q = \boldsymbol{\eta}_L + \phi(\boldsymbol{\lambda}) + \psi(\boldsymbol{\Lambda})$, where ϕ and ψ are linear operators (see App. F.3).

Inference In this scenario, the ELBO is

$$\mathcal{L}(q) = -\mathrm{D}_{\mathrm{KL}}[q(\{\mathbf{x}\}_\tau \,\|\, p_L(\{\mathbf{x}\}_\tau)] \qquad (21)$$
$$+ \sum_{i=1}^{n} \mathbb{E}_{q(\mathbf{x}_i^{\mathrm{d}})} \log p(y_i \mid \mathbf{x}_i^{\mathrm{d}}) + \sum_{m=0}^{M-1} \mathbb{E}_{q(\tilde{\mathbf{x}}_m)} V(\tilde{\mathbf{x}}_m)\,\Delta t\,.$$

The inference algorithm consists of applying Mirror descent on this objective in the expectation parameterization, which leads to the following *local* updates

$$\boldsymbol{\lambda}_i^{(k+1)} = (1-\rho)\boldsymbol{\lambda}_i^{(k)} + \rho\phi_i^{-1}\big(\nabla_{\boldsymbol{\mu}_i}\mathbb{E}_{q^{(k)}}\log p(y_i \mid \mathbf{x}_i^{\mathrm{d}})\big), \ (22)$$
$$\boldsymbol{\Lambda}_m^{(k+1)} = (1-\rho)\boldsymbol{\Lambda}_m^{(k)} + \rho\,\psi_m^{-1}\big(\nabla_{\tilde{\boldsymbol{\mu}}_m}\mathbb{E}_{q^{(k)}}V(\tilde{\mathbf{x}}_m)\big), \qquad (23)$$

where ϕ_i and ψ_m are additional linear operators (see App. F.4). It is important to note that the choice of p_L is not important for the purpose of inference. This is because q^* and $\boldsymbol{\lambda}^*$ are independent of p_L. For any distribution p_L, the optimal dense sites are given by $\psi(\boldsymbol{\Lambda}^*) = \boldsymbol{\eta}_{q^*} - \boldsymbol{\eta}_L - \phi(\boldsymbol{\lambda}^*)$. However, for the purpose of learning, the choice of p_L matters.

Learning For the purpose of learning the hyperparameters $\boldsymbol{\theta}$ of the prior diffusion, we use the modified VEM approach described in App. E.1. For this method to be effective, it is desirable at the M-step for the sites $\boldsymbol{\lambda}, \boldsymbol{\Lambda}$ of the variational distribution to be as independent of $\boldsymbol{\theta}$ as possible, or alternatively to have p_L as close as possible to p. Here, we use posterior linearization (Eq. (10)) and set $(\mathbf{A}_m, \mathbf{b}_m) = \Pi_{q(\mathbf{x}_m)}[f_p(\cdot, t_m)]$. When the prior diffusion p is linear, we recover exactly the algorithm for linear diffusion discussed in Sec. 3.

Table 1: Negative log predictive density (NLPD) (lower better) over 5-fold CV for Gaussian (sanity check) and non-Gaussian DPs when performing **inference** only. CVI-DP agrees with the SMC ground-truth, even at large discretization steps Δt while VDP suffers from convergence issues.

		Gaussian OU	Beneš	DW	Non-Gaussian priors Sine	Sqrt	van der Pol
Baseline	SMC	0.371 ± 0.225	0.178 ± 0.160	0.069 ± 0.140	0.153 ± 0.210	-0.054 ± 0.085	0.836 ± 0.460
	Opt. Gaussian	0.374 ± 0.234	0.172 ± 0.167	0.077 ± 0.117	0.160 ± 0.215	-0.046 ± 0.086	0.946 ± 0.711
	GPR (OU kernel)	0.377 ± 0.241	0.191 ± 0.130	0.252 ± 0.091	0.220 ± 0.281	0.648 ± 1.371	0.638 ± 0.447
Δt 0.01	VDP	1.842 ± 2.494	0.253 ± 0.128	0.412 ± 0.231	0.208 ± 0.169	0.162 ± 0.155	0.042 ± 0.609
	CVI-DP (Ours)	0.377 ± 0.239	0.169 ± 0.153	0.076 ± 0.186	0.154 ± 0.207	-0.048 ± 0.078	-0.144 ± 0.534
Δt 0.005	VDP	0.379 ± 0.241	0.174 ± 0.101	0.384 ± 0.215	0.160 ± 0.191	0.070 ± 0.146	-0.150 ± 0.431
	CVI-DP (Ours)	0.377 ± 0.240	0.169 ± 0.153	0.075 ± 0.190	0.154 ± 0.207	-0.050 ± 0.078	-0.159 ± 0.514
Δt 0.001	VDP	0.377 ± 0.240	0.154 ± 0.126	0.318 ± 0.228	0.167 ± 0.193	0.059 ± 0.144	-0.148 ± 0.434
	CVI-DP (Ours)	0.377 ± 0.241	0.169 ± 0.153	0.075 ± 0.192	0.154 ± 0.208	-0.050 ± 0.078	-0.167 ± 0.501

Algorithm 1: Variational Inference under a discretized non-linear DP prior.

Input: Prior $p(\{\mathbf{x}\}_\tau), q(\{\mathbf{x}\}_\tau)$, data $\mathcal{D}$, learning rate ρ
while *not converged* **do**
    while *not converged* **do**
        Sites update:
        $\boldsymbol{\lambda}_i^{(k+1)} = (1-\rho)\boldsymbol{\lambda}_i^{(k)} + \rho\,\phi_i^{-1}(\nabla_{\boldsymbol{\mu}_i}\mathbb{E}_{q^{(k)}}\log p(y_i\,|\,\mathbf{x}_i^{\mathrm{d}}))$
        $\boldsymbol{\Lambda}_m^{(k+1)} = (1-\rho)\boldsymbol{\Lambda}_m^{(k)} + \rho\,\psi_m^{-1}(\nabla_{\tilde{\boldsymbol{\mu}}_m}\mathbb{E}_{q^{(k)}}V(\tilde{\mathbf{x}}_m))$
        Compute posterior:
        $\boldsymbol{\eta}_q^{(k+1)} = \boldsymbol{\eta}_{p_L} + \phi(\boldsymbol{\lambda}^{(k+1)}) + \psi(\boldsymbol{\Lambda}^{(k+1)})$
    end
    Update p_L via posterior linearization:
    $(\mathbf{A}_m, \mathbf{b}_m)^{new} \leftarrow \Pi_{q(\mathbf{x}_m)}[f_p(\cdot, t_m)]$
    Update $\boldsymbol{\Lambda}$ under p_L^{new}: $\boldsymbol{\Lambda}|_{p_L^{\mathrm{new}}} = \boldsymbol{\Lambda}|_{p_L} + \boldsymbol{\eta}_{p_L} - \boldsymbol{\eta}_{p_L^{\mathrm{new}}}$
end

We choose to set p_L iteratively via posterior linearization within a broader CVI-DP algorithm for inference and learning, which we describe in Alg. 1. We alternate steps of *(i)* inference via mirror descent on the ELBO (Eq. (22) and Eq. (23)), *(ii)* posterior linearization to update p_L, and *(iii)* learning via gradient descent on an ELBO with respect to $\boldsymbol{\theta}$.

4.3 Continuous-time Limit

When taking the continuous-time limit $\mathrm{d}t \to 0$, both p_L and q become Gaussian processes, and these no longer have a formulation through density functions with respect to the Lebesgue measure. As $\mathrm{d}t \to 0$, Girsanov theorem guarantees the existence of the change of measure in Eq. (17), and we have

$$\mathbb{E}_{q(\{\mathbf{x}\}_\tau)}\log\frac{p(\{\mathbf{x}\}_\tau)}{p_L(\{\mathbf{x}\}_\tau)} \underset{\mathrm{d}t\to0}{=} \mathbb{E}_{\mathsf{Q}}\log\frac{\mathsf{P}(\mathrm{d}\mathbf{x})}{\mathsf{P}_L(\mathrm{d}\mathbf{x})}$$
$$= \int \mathbb{E}_q\|f_L(\mathbf{x}_t, t) - f_p(\mathbf{x}_t, t)\|^2_{\mathbf{Q}_{\mathrm{c}}^{-1}}\mathrm{d}t, \quad (24)$$

where $\mathsf{P}, \mathsf{P}_L, \mathsf{Q}$ are the probability measures of the limiting processes p, p_L, q respectively. Informally, as

the sites get dense, sums become integrals

$$q(\{\mathbf{x}\}_\tau) \underset{\mathrm{d}t\to0}{\propto} p_L(\mathbf{x})\exp\Big(\sum_{i=1}^n\langle\phi_i(\boldsymbol{\lambda}_i), \mathsf{T}(\mathbf{x}_i^{\mathrm{d}})\rangle$$
$$+ \int_0^T\langle\boldsymbol{\Lambda}_t, \mathsf{T}(\mathbf{x}_t)\rangle\,\mathrm{d}t\Big). \quad (25)$$

Though such a density does not exist, the marginal means $\mathbf{m}_t$ and covariances $\boldsymbol{\Sigma}_t$ of q do exist and are solution to the well-defined ODEs of the Kalman–Bucy filter (Särkkä and Svensson, 2023). Also, the local site update rules Eq. (22) and Eq. (23) readily extend to the continuous time setting

$$\boldsymbol{\Lambda}_t^{(k+1)} = (1-\rho)\boldsymbol{\Lambda}_t^{(k)} + \rho\,\psi_t^{-1}\big(\nabla_{\boldsymbol{\mu}_t}\mathbb{E}_{q^{(k)}}V(\mathbf{x}_t)\big). \quad (26)$$

We provide further details in App. G. It is important to note that, after taking the continuous time limit, the algorithm no longer depends on the Euler–Maruyama discretization used to build the discrete algorithm in the first place.

Relation to the original CVI algorithm CVI was introduced in Khan and Lin (2017) with application to inference in Gaussian process models. It was extended to the case of Markovian Gaussian processes in Chang et al. (2020) and to the setting of sparse Gaussian processes (Wilkinson et al., 2021; Adam et al., 2021), an approximation framework to reduce the computational cost of inference. The modified VEM algorithm was explicitly introduced in Adam et al. (2021). In this paper, we further extended CVI to the setting of non-linear diffusions.

Comparison with the VDP method CVI-DP is built on top of efficient algorithms specialized for the setting of linear diffusion priors (*i.e.*, GPs). One of its good properties is that it reverts to these efficient pre-existing algorithms when applied to problems in the linear-Gaussian setting. In that sense, it unifies variational inference in models with a GP and a DP prior. This is unlike the VDP, which is very inefficient in these scenarios in terms of speed of convergence for both inference and learning (*cf.* Figs. 3 and 5).

Table 2: Negative log predictive density (NLPD) (lower better) over 5-fold CV for Gaussian (sanity check) and non-Gaussian DPs when performing **inference and learning**. CVI-DP yields good performance even at large discretization steps Δt while VDP suffers from convergence issues.

		Gaussian OU	Beneš	DW	Non-Gaussian priors Sine	Sqrt	van der Pol
	GPR (OU kernel)	$0.389{\pm}0.249$	$0.197{\pm}0.164$	$0.218{\pm}0.091$	$0.214{\pm}0.202$	$0.358{\pm}0.134$	$0.561{\pm}0.154$
Δt 0.01	VDP	$0.559{\pm}0.488$	$0.642{\pm}0.577$	$0.270{\pm}0.142$	$0.209{\pm}0.169$	$0.170{\pm}0.152$	$0.713{\pm}0.827$
	CVI-DP (Ours)	$0.361{\pm}0.240$	$0.154{\pm}0.146$	$0.100{\pm}0.131$	$0.180{\pm}0.218$	$-0.048{\pm}0.083$	$-0.032{\pm}0.412$
Δt 0.005	VDP	$0.385{\pm}0.233$	$0.218{\pm}0.184$	$0.223{\pm}0.149$	$0.162{\pm}0.183$	$0.085{\pm}0.141$	$0.192{\pm}0.577$
	CVI-DP (Ours)	$0.361{\pm}0.239$	$0.154{\pm}0.146$	$0.098{\pm}0.134$	$0.177{\pm}0.214$	$-0.052{\pm}0.079$	$-0.031{\pm}0.412$
Δt 0.001	VDP	$0.374{\pm}0.239$	$0.209{\pm}0.191$	$0.297{\pm}0.185$	$0.165{\pm}0.197$	$0.028{\pm}0.133$	$-0.032{\pm}0.426$
	CVI-DP (Ours)	$0.361{\pm}0.239$	$0.154{\pm}0.147$	$0.100{\pm}0.135$	$0.177{\pm}0.213$	$-0.044{\pm}0.081$	$-0.022{\pm}0.404$

However, the parameterization of CVI-DP is slightly more costly than that of VDP. Both methods implicitly learn the transition statistics $(\hat{\mathbf{A}}_i, \hat{\mathbf{b}}_i, \hat{\mathbf{Q}}_i)$ of a linear Gaussian SSM (see Eq. (13)), but VDP does not learn the $\hat{\mathbf{Q}}_i$ and restricts the diffusion coefficient to match that of the prior diffusion. This is suboptimal when the continuous algorithm is discretized at implementation time: in Fig. 3 we show that the performance of VDP degrades fast as the discretization gets coarser, while the performance of CVI-DP is relatively unaffected.

5 EXPERIMENTS

We implement both CVI-DP and VDP within the MarkovFlow (Adam et al., 2022) framework built on top of TensorFlow (Abadi et al., 2015) and perform a series of experiments to showcase various properties of these methods. For inference and learning, we evaluate the performance of these methods on problems covering a set of DP priors, both linear and non-linear, each with its own characteristics. For inference, we show the proposed method CVI-DP performs better than VDP and is at par with the sequential Monte Carlo baseline. For learning, we use the same setup but also learn the model parameters of the DP prior. We show how CVI-DP provides a better learning objective leading to faster learning. Furthermore, to showcase the applicability of CVI-DP in the real world, we demonstrate inference on finance and GPS tracking data sets.

Synthetic problems We comparatively evaluate our method on synthetic problems covering an array of DP priors: the linear DP (Ornstein–Uhlenbeck, OU, Fig. A8), $dx_t = -\theta x_t\, dt + d\beta_t$, as a sanity check for which the posterior process can be written in closed-form; the Beneš DP, $dx_t = \theta \tanh(x_t)\, dt + d\beta_t$, whose marginal state distributions are bimodal and mode-switching in sample state trajectories becomes increasingly unlikely with time (Fig. A9); the Double-Well (DW) DP, $dx_t = \theta_0 x_t(\theta_1 - x_t^2)\, dt + d\beta_t$, whose marginal state distributions have two modes that sample state trajectories keep visiting through time (Fig. 3); a Sine DP, $dx_t = \theta_0 \sin(x_t - \theta_1)\, dt + d\beta_t$,

whose marginal state distributions have many modes (Fig. A10); a Square-root DP, $dx_t = \sqrt{\theta|x_t|}\, dt + d\beta_t$, that has divergent fat-tailed behaviour (Fig. A11); and stochastic van der Pol oscillator, $d\mathbf{x}_t = \theta_0(\theta_1 x_t^{(1)} - (1/3)x_t^{(1)} - x_t^{(2)})\, dt, (1/\theta_1)x_t^{(1)}\, dt)^\top + d\boldsymbol{\beta}_t$, that has multi-dimensional state vector (Fig. 4).

Baselines As a baseline, we use sequential Monte Carlo (SMC) as particle smoothing through conditional particle filtering with ancestor sampling adopted from Svensson et al. (2015). The 'optimal' Gaussian baseline is based on a Gaussian fit to the SMC samples. To approximate the log marginal likelihood, we use annealed importance sampling (AIS, Neal, 2001) with a similar setup as in Kuss and Rasmussen (2005); Nickisch and Rasmussen (2008) (details in App. H). We maintain a consistent number of particles and smoothers across all experiments. However, it struggles with convergence issues in the multi-dimensional van der Pol DP.

Speeding up inference We compare the performance of approximate inference using VDP and CVI-DP on three aspects: convergence speed, accuracy of the posterior approximation, and robustness to the discretization of the time horizon. As a sanity check, we start with a linear DP prior, where CVI-DP reaches the optimal posterior after a single iteration (Fig. A8). Also, for non-linear DP priors, CVI-DP is faster than VDP (see Fig. 3 for an example with DW prior). Convergence plots for other DPs are available in App. I. To measure the accuracy of the approximate posterior, we use negative log predictive density (NLPD) with 5-fold cross-validation (results in Table 1). The proposed method CVI-DP is robust to the discretization of the time horizon; we report both the methods with different discretization ($\Delta t = \{0.01, 0.005, 0.001\}$). A coarse grid leads to a model with fewer parameters as the number of variational parameters scales inversely with Δt.

Learning of model parameters To showcase the learning capability of CVI-DP, we experiment with the same setup as in the evaluation of inference but now learn parameters of the prior DP $\boldsymbol{\theta}$ as well. We compare the two methods on two aspects: speed of learning and
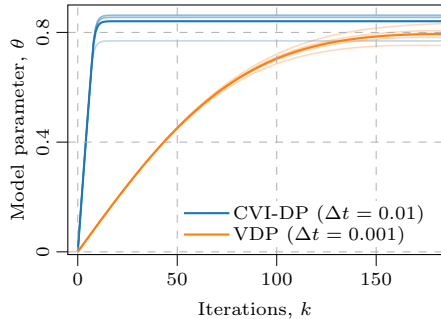
Figure 5: Faster learning of the Double-Well DP parameter θ (M-Step) of the proposed method CVI-DP compared to VDP.
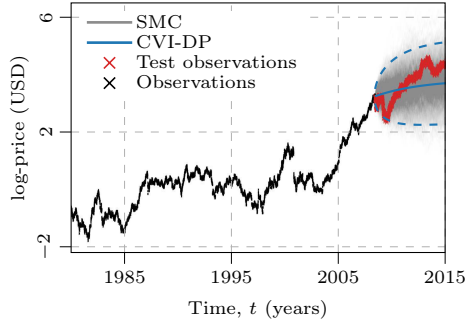
Figure 6: Apple Inc. stock price data set. CVI-DP learns the drift as an MLP neural net and predict into the future.
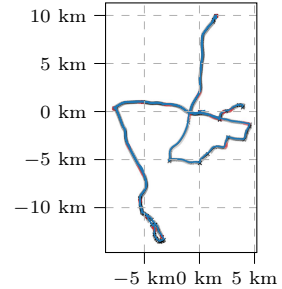
Figure 7: CVI-DP with NN drift on GPS data along with observations.

posterior predictive accuracy. CVI-DP provides a better objective for learning (Fig. 2) which leads to a faster learning algorithm (Fig. 5). We also report the posterior predictive performance of the methods in Table 2 using NLPD with 5-fold cross-validation as a metric.

Finance data We model the trend of Apple Inc. share price (8537 trading days), and evaluate four models under 5-fold cross-validation: Sparse GP regression model with 500 inducing points (Titsias, 2009) and a sum kernel of Const.+Lin.+Matérn-1/2+Matérn-3/2 as in Solin and Särkkä (2015) which gives NLPD 1.44±0.70/RMSE 0.91±0.54; CVI-DP with a linear DP (OU) which gives NLPD 1.08±0.45/RMSE 0.77±0.41; CVI-DP with a neural network drift DP which gives NLPD 0.81±0.08/RMSE 0.51±0.08; and VDP model with a neural network drift DP gives NLPD 0.92±0.22/RMSE 0.54±0.17. The models have different priors incorporated in the form of the kernel and prior DP, giving different flexibility to the resulting variational posterior. The aim is to learn the underlying process of the stock price, which we measure in terms of NLPD on the hold-out set. Of all the models, CVI-DP with an NN drift results in the best NLPD value due to its flexibility. Fig. 6 shows simulated predictions from the CVI-DP learnt prior DP (details in App. I.2).

Vehicle tracking We model a 2D trajectory of vehicle movement from GPS coordinates (Fig. 7). The data set consists of 6373 observation points collected over 106 minutes. Outputs are modelled with two independent DPs learned jointly. Similar to the setup in Solin and Särkkä (2015), we split the data in chunks of 30 s and perform 10-fold cross-validation. We experiment with three models: CVI-DP with a linear DP (OU), which gives NLPD -0.67±0.19/RMSE 0.13±0.04; CVI-DP with a NN drift DP which gives NLPD -0.82±0.43/RMSE 0.06±0.03; and VDP with a NN drift DP which gives NLPD -0.55±0.24/RMSE 0.09±0.06. CVI-DP with a NN drift gives better NLPD and RMSE value primarily because of the flexibility of the DP to model the trajectory (details in App. I.3).

Further comparisons Finally, we compare against the popular class of NeuralSDEs methods (Li et al., 2020a; Kidger et al., 2021). These methods are variational inference algorithms with a broader scope than CVI-DP: the posterior process q is not restricted to be a linear DP. However, they rely on sample-based estimation of the ELBO gradient and thus incur a large computational cost, and convergence of optimization via stochastic gradient descent is often slow (see App. I.4).

6 DISCUSSION AND CONCLUSION

CVI-DP provides a principled and scalable approach for variational inference in models with latent non-linear diffusion processes, parameterized via a time-variant linear Itô SDE. We argue that the Gaussian VI approach commonly used for inference in GP models can readily be extended to the DP prior setting, and propose a unifying approach for both the GP and DP prior settings. Our work fixes practical problems in the seminal work by Archambeau et al. (2007a) (*cf.* Figs. 2 and 3), and provides an important building block for approximative inference and learning under diffusion processes.

While we tackle the core SDE tooling, diffusion processes have recently gained traction in fields across machine learning such as image generation (Ho et al., 2020; Song et al., 2021; Dhariwal and Nichol, 2021), reinforcement learning (Janner et al., 2022), and time-series modelling (Vargas et al., 2021; Tashiro et al., 2021; Park et al., 2022). We see that our work can be an important component in future methods and extended to handle more complex scenarios with additional constraints or priors.

A reference implementation of the methods presented in this paper is available at https://github.com/AaltoML/vi-diffusion-processes.

Acknowledgements

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/.

V. Adam, P. Chang, M. E. Khan, and A. Solin. Dual parameterization of sparse variational Gaussian processes. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pages 11474–11486. Curran Associates, Inc., 2021.

V. Adam, A. Artemev, N. Durrande, S. Eleftheriadis, A. Hayes, J. Hensman, J. Jennings, S. John, J. A. Leedham, J. A. McLeod, A. Saul, P. Verma, Y. Wei, and S. Willis. Markovflow, 2022. URL https://github.com/secondmind-labs/markovflow.

S. I. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

C. Andrieu, A. Doucet, and R. Holenstein. Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342, 2010.

C. Archambeau and M. Opper. *Approximate Inference for Continuous-Time Markov Processes*, pages 125–140. Cambridge University Press, 2011.

C. Archambeau, D. Cornford, M. Opper, and J. Shawe-Taylor. Gaussian process approximations of stochastic differential equations. In *Gaussian Processes in Practice*, volume 1 of *Proceedings of Machine Learning Research*, pages 1–16. PMLR, 2007a.

C. Archambeau, M. Opper, Y. Shen, D. Cornford, and J. Shawe-taylor. Variational inference for diffusion processes. In *Advances in Neural Information Processing Systems (NIPS)*, pages 17–24. Curran Associates, Inc., 2007b.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518): 859–877, 2017.

M. Y. Byron, K. V. Shenoy, and M. Sahani. Derivation of Kalman filtering and smoothing equations. Technical report, Stanford University, 2004.

E. Challis and D. Barber. Gaussian Kullback-Leibler approximate inference. *Journal of Machine Learning Research*, 14:2239–2286, 2013.

P. E. Chang, W. J. Wilkinson, M. E. Khan, and A. Solin. Fast variational learning in state-space Gaussian process models. In *Proceedings of the 30th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6. IEEE, 2020.

R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pages 6571–6583. Curran Associates, Inc., 2018.

N. Chopin, O. Papaspiliopoulos, et al. *An Introduction to Sequential Monte Carlo*. Springer, 2020.

K. Course and P. B. Nair. State estimation of a physical system with unknown governing equations. *Nature*, 622(7982):261–267, 2023.

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

B. Cseke, M. Opper, and G. Sanguinetti. Approximate inference in latent Gaussian-Markov models from continuous time observations. 2013.

B. Cseke, D. Schnoerr, M. Opper, and G. Sanguinetti. Expectation propagation for continuous time stochastic processes. *Journal of Physics A: Mathematical and Theoretical*, 49(49):494002, 2016.

P. Dhariwal and A. Q. Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, pages 8780–8794. Curran Associates, Inc., 2021.

L. Duncker, G. Bohner, J. Boussard, and M. Sahani. Learning interpretable continuous-time models of latent stochastic dynamical systems. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pages 1726–1734. PMLR, 2019.

B. Eraker. MCMC analysis of diffusion models with application to finance. *Journal of Business & Economic Statistics*, 19:177–191, 2001.

C. A. García, A. Otero, P. Félix, J. Presedo, and D. G. Márquez. Nonparametric estimation of stochastic

differential equations with sparse Gaussian processes. *Physical Review E*, 96, 2017.

Á. F. García-Fernández, L. Svensson, M. R. Morelande, and S. Särkkä. Posterior linearization filter: Principles and implementation using sigma points. *IEEE Transactions on Signal Processing*, 63(20):5561–5573, 2015.

Á. F. García-Fernández, L. Svensson, and S. Särkkä. Iterated posterior linearization smoother. *IEEE Transactions on Automatic Control*, 62(4):2056–2063, 2016.

Á. F. García-Fernández, F. Tronarp, and S. Särkkä. Gaussian process classification using posterior linearization. *IEEE Signal Processing Letters*, 26(5): 735–739, 2019.

I. Girsanov. On transforming a certain class of stochastic processes by absolutely continuous substitution of measures. *Theory of Probability and Its Applications*, 5:314–330, 1960.

A. Golightly and D. J. Wilkinson. Bayesian parameter inference for stochastic biochemical network models using particle Markov chain Monte carlo. *Interface Focus*, 1:807–820, 2011.

M. C. Higgs. *Approximate Inference for State-Space Models*. PhD thesis, University College London, London, UK, 2011.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pages 6840–6851. Curran Associates, Inc., 2020.

M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, and S. Sabato, editors, *International Conference on Machine Learning (ICML)*, volume 162, pages 9902–9915. PMLR, 2022.

I. Karatzas and S. E. Shreve. Brownian motion. In *Brownian Motion and Stochastic Calculus*, pages 47–127. Springer, 1998.

M. E. Khan and W. Lin. Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, pages 878–887. PMLR, 2017.

P. Kidger, J. Foster, X. C. Li, and T. Lyons. Efficient and accurate gradients for neural SDEs. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pages 18747–18761. Curran Associates, Inc., 2021.

L. Köhs, B. Alt, and H. Koeppl. Variational inference for continuous-time switching dynamical systems. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pages 20545–20557. Curran Associates, Inc., 2021.

M. Kuss and C. E. Rasmussen. Assessing approximate inference for binary Gaussian process classification. *Journal of Machine Learning Research (JMLR)*, 6: 1679–1704, 2005.

X. Li, T.-K. L. Wong, R. T. Q. Chen, and D. Duvenaud. Scalable gradients for stochastic differential equations. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 3870–3882. PMLR, 2020a.

X. Li, T.-K. L. Wong, R. T. Q. Chen, and D. K. Duvenaud. Scalable gradients and variational inference for stochastic differential equations. In *Proceedings of The 2nd Symposium on Advances in Approximate Bayesian Inference*, volume 118 of *Proceedings of Machine Learning Research*, pages 1–28. PMLR, 2020b.

W. Lin, M. E. Khan, and M. Schmidt. Fast and simple natural-gradient variational inference with mixture of exponential-family approximations. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pages 3992–4002. PMLR, 2019.

F. Lindsten, M. I. Jordan, and T. B. Schon. Particle Gibbs with ancestor sampling. *Journal of Machine Learning Research (JMLR)*, 15:2145–2184, 2014.

T. P. Minka. Expectation propagation for approximate Bayesian inference. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 362–369. Morgan Kaufmann, 2001.

R. M. Neal. Annealed importance sampling. *Statistics and Computing*, 11:125–139, 2001.

R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, pages 355–368. Springer, 1998.

H. Nickisch and C. E. Rasmussen. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research (JMLR)*, 9:2035–2078, 2008.

B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, New York, NY, sixth edition, 2003.

S. W. Park, K. Lee, and J. Kwon. Neural Markov controlled SDE: Stochastic optimization for continuous-time data. In *International Conference on Learning Representations (ICLR)*, 2022.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Y. Rubanova, R. T. Chen, and D. K. Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, pages 5320–5330. Curran Associates, Inc., 2019.

A. Ruttor, P. Batz, and M. Opper. Approximate Gaussian process inference for the drift function in stochastic differential equations. In *Advances in Neural Information Processing Systems 26 (NIPS)*, pages 2040–2048. Curran Associates, Inc., 2013.

T. Ryder, A. Golightly, A. S. McGough, and D. Prangle. Black-box variational inference for stochastic differential equations. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 4423–4432. PMLR, 2018.

S. Särkkä and A. Solin. *Applied Stochastic Differential Equations*. Cambridge University Press, 2019.

M.-A. Sato. Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681, 2001.

A. Solin and S. Särkkä. State space methods for efficient inference in Student-t process regression. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 38 of *Proceedings of Machine Learning Research*, pages 885–893. PMLR, 2015.

Y. Song, C. Durkan, I. Murray, and S. Ermon. Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, pages 1415–1428. Curran Associates, Inc., 2021.

A. Svensson, T. B. Schön, and M. Kok. Nonlinear state space smoothing using the conditional particle filter. In *Proceedings of the 17th IFAC Symposium on System Identification (SYSID)*, volume 48, pages 975–980. Elsevier, 2015.

S. Särkkä and L. Svensson. *Bayesian Filtering and Smoothing*. Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2 edition, 2023.

Y. Tashiro, J. Song, Y. Song, and S. Ermon. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, pages 24804–24816. Curran Associates, Inc., 2021.

M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574. PMLR, 2009.

F. Tronarp, A. F. Garcia-Fernandez, and S. Särkkä. Iterative filtering and smoothing in nonlinear and non-Gaussian systems using conditional moments. *IEEE Signal Processing Letters*, 25(3):408–412, 2018.

N. G. van Kampen. *Stochastic Processes in Physics and Chemistry*, volume 1. Elsevier, 1992.

F. Vargas, P. Thodoroff, A. Lamacraft, and N. Lawrence. Solving Schrödinger bridges via maximum likelihood. *Entropy*, 23(9):1134, 2021.

N. Whiteley. Discussion on particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B*, 72(3):306–307, 2010.

C. Wildner and H. Koeppl. Moment-based variational inference for stochastic differential equations. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research, pages 1918–1926. PMLR, 2021.

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

W. J. Wilkinson, S. Särkkä, and A. Solin. Bayes–Newton methods for approximate Bayesian inference with PSD guarantees. *Journal of Machine Learning Research (JMLR)*, 24(83):1–50, 2023.

C. Yildiz, M. Heinonen, J. Intosalmi, H. Mannerstrom, and H. Lahdesmaki. Learning stochastic differential equations with Gaussian processes without gradient matching. In *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2018.

Checklist

1. For all models and algorithms presented, check if you include:

   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]

   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]

   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:

   (a) Statements of the full set of assumptions of all theoretical results. [Yes]

   (b) Complete proofs of all theoretical results. [Yes]

   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:

   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]

   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]

   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]

   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

   (a) Citations of the creator If your work uses existing assets. [Yes]

   (b) The license information of the assets, if applicable. [Yes]

   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]

   (d) Information about consent from data providers/curators. [Not Applicable]

   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. [Not Applicable]

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Supplementary Material: Variational Gaussian Process Diffusion Processes

This supplementary document is organized as follows. App. A describes exact inference for latent diffusion models, and highlights the properties of the posterior drift function. App. B describes an (intractable) variational algorithm to perform exact inference in latent diffusion models, using the method of Lagrangian multipliers to optimize a constrained objective. App. C provides the full derivations for the tractable variational algorithm of Archambeau et al. (2007a), where the variational distribution is restricted to a set of Markovian Gaussian Process. In App. D, we derive linear Gaussian state space model using an alternative parameterization. In App. E, we derive the proposed method CVI-DP (discrete version) for linear DP prior and an arbitrary observation model. In App. F, we derive the proposed method CVI-DP (discrete version) for non-linear DP prior and an arbitrary observation model. In App. G, we derive the proposed method CVI-DP (continuous version) for non-linear DPs. App. H gives details about the Monte Carlo baselines and App. I provides details about the experiment setup and insights about various results.

A   Exact Inference for Latent Diffusion Process Models

In this section, we describe the exact posterior inference in models with diffusion process prior, following the derivations in Higgs (2011). Given the Markovian structure of the diffusion, the marginal posterior at time t factorizes as

$$p(\mathbf{x}_t \,|\, \mathcal{D}) = \underbrace{p(\mathbf{x}_t \,|\, \mathcal{D}_{<t})}_{p_t^{(F)}(\mathbf{x}_t)} \underbrace{\frac{p(\mathcal{D}_{\geq t} \,|\, \mathbf{x}_t)}{p(\mathcal{D}_{\geq t} \,|\, \mathcal{D}_{<t})}}_{\psi_t(\mathbf{x}_t)}. \tag{A27}$$

This expression splits the contribution of observations before and after time t into two terms corresponding to continuous time equivalent of the forward and backward (up to a constant) filtering distributions in discrete Markov chains (Byron et al., 2004).

The forward filtering distribution $p_t^{(F)}$ satisfies, in between observation times, the Kolmogorov forward equation (Karatzas and Shreve, 1998)

$$(\partial_t - \overrightarrow{K}_f)p_t^{(F)} = 0, \tag{A28}$$

where $\overrightarrow{K}_f$ is the Fokker–Planck operator defined for twice differentiable functions ϕ as

$$\overrightarrow{K}_f[\phi(\mathbf{x})] := -\nabla^\top[f(\mathbf{x}, t)\phi(\mathbf{x})] + \frac{1}{2}\mathrm{tr}(\mathbf{Q}_c(\nabla\nabla^\top)[\phi(\mathbf{x})]). \tag{A29}$$

Similarly, the second term $\psi_t(\mathbf{x}_t)$—also known as the information filter—satisfies, in between observation times, the Kolmogorov backward equation

$$(\partial_t + \overleftarrow{K}_f)\psi_t = 0, \tag{A30}$$

where $\overleftarrow{K}_f$ is the adjoint of the Fokker–Planck operator defined for twice differentiable functions ϕ as

$$\overleftarrow{K}_f[\phi(\mathbf{x})] := f(\mathbf{x}, t)^\top \nabla[\phi(\mathbf{x})] + \frac{1}{2}\mathrm{tr}(\mathbf{Q}_c(\nabla\nabla^\top)[\phi(\mathbf{x})]). \tag{A31}$$

If solving for $p_t^{(F)}$ forward in time, at the discrete observation times t, observations are added from the conditioning sets of $p_t(F)$, leading to the instantaneous updates

$$p_{t+}^{(F)}(\mathbf{x}_{t+}) \propto p_{t-}^{(F)}(\mathbf{x}_{t-}) \, p(\mathcal{D}_t \,|\, \mathbf{x}_{t-}). \tag{A32}$$

The information filter ψ_t can be solved backwards with the initial condition $\psi_T = 1$ and discrete updates

$$\psi_{t-}(\mathbf{x}_{t-}) = \frac{p(\mathcal{D}_t \,|\, \mathbf{x}_t)\,\psi_{t+}(\mathbf{x}_{t+})}{p(\mathcal{D}_t \,|\, \mathcal{D}_{<t})}. \tag{A33}$$

Differentiating the marginal posterior distribution $p_{\mathbf{x}_t \,|\, \mathcal{D}}$ through time (Higgs, 2011, Theorem 2.8), it can be shown that the posterior process $p_{\mathbf{x} \,|\, \mathbf{y}}$ shares the same diffusion as the prior process and its drift h is given by:

$$h(\mathbf{x}_t, t) = f_p(\mathbf{x}_t, t) + \mathbf{Q}_{\mathrm{c}} \nabla \log \psi_t(\mathbf{x}_t). \tag{A34}$$

This result reveals that the posterior process shares the same Markovian structure as the prior process. It also provides a recipe to compute the posterior marginals: (1) compute ψ_t backward in time via Eq. (A30) and the jump conditions Eq. (A33), and (2) compute the posterior drift via Eq. (A34) and compute the marginals of the posterior process via Eq. (A28). These steps are however intractable for most settings of non-linear drift and observation models.

B Variational Inference for Latent Diffusion Process Models (General Case)

In this section, we derive the posterior process for latent diffusion models starting from the variational formulation of the problem. We introduce the variational objective, describe an optimization procedure and discuss some properties of the solution.

We start from the variational objective for inference in latent diffusion models given in Eq. (5), which we rewrite here for completeness

$$\mathcal{L}(q) = \mathbb{E}_{q(\mathbf{x})}[\log p(\mathbf{y} \,|\, \mathbf{x})] - \frac{1}{2} \int_0^\tau \mathbb{E}_{q(\mathbf{x}_t)} \| f_q(\mathbf{x}_t, t) - f_p(\mathbf{x}_t, t) \|_{\mathbf{Q}_{\mathrm{c}}^{-1}}^2 \, dt - \mathrm{D}_{\mathrm{KL}}[q(\mathbf{x}_0) \,\|\, p(\mathbf{x}_0)]. \tag{A35}$$

The drift f_{q^*} of the optimal variational process q^* maximizing $\mathcal{L}(q)$ can be derived from this variational objective using the tools of Lagrangian duality. Let $\tilde{\mathcal{L}}(q, f_q)$ be the ELBO defined in Eq. (A35) where we have severed the dependency between q and f_q. The optimization of the ELBO can be cast as the optimization problem

$$\max_q \tilde{\mathcal{L}}(q, f_q) \quad \text{subject to} \quad \big[(\partial_t - \overrightarrow{K}_{f_q})q\big](\mathbf{x}, t) = 0, \quad \forall \mathbf{x}, t, \tag{A36}$$

where $\overrightarrow{K}_f$ is the Fokker–Planck operator defined in Eq. (A29). We introduce the Lagrangian λ associated to this constrained optimization problem

$$\mathfrak{L}(q, f_q, \lambda) = \tilde{\mathcal{L}}(q, f_q) + \int \int \lambda(\mathbf{x}_t, t) \big[(\partial_t - \overrightarrow{K}_{f_q})q\big](\mathbf{x}_t, t) \, d\mathbf{x}_t \, dt. \tag{A37}$$

A local optimum (q^*, f_q^*) of the original constrained optimization problem is associated to a unique pair of Lagrangian multipliers λ^* satisfying $\nabla_{q, f_q, \lambda} \mathfrak{L}|_{q^*, f_q^*, \lambda^*} = \mathbf{0}$. Introducing the change of variable $\lambda(\mathbf{x}, t) = -\log \psi(\mathbf{x}, t)$, the system of equations can be written as

$$0 = \frac{\partial \psi(\mathbf{x}, t)}{\partial t} + \overleftarrow{K}_f[\psi(\mathbf{x}, t)] + \psi(\mathbf{x}, t) \sum_{i=1}^n [\log p(y_i \,|\, \mathbf{x}_i)] \delta(t - t_i). \tag{A38}$$

We recover the posterior drift as in Eq. (A34),

$$f_q(\mathbf{x}_t, t) = f_p(\mathbf{x}_t, t) + \mathbf{Q}_{\mathrm{c}} \nabla \log \psi(\mathbf{x}_t, t). \tag{A39}$$

C Gaussian Variational Inference for Diffusion Processes (VDP)

In this section, we focus on the case where the set of variational processes is restricted to that of Markovian Gaussian processes, as described in Sec. 2.1. More formally, the variational process q is restricted to be a diffusion with an affine drift

$$\mathcal{Q} = \{q \,:\, d\mathbf{x}_t = (\mathbf{A}_t \mathbf{x}_t + \mathbf{b}_t)\, dt + \mathbf{L}\, d\boldsymbol{\beta}_t, \quad \mathbf{x}_0 \sim q(\mathbf{x}_0)\}. \tag{A40}$$

The drift parameters $\mathbf{A}_t \in \mathbb{R}^{d \times d}$ and $\mathbf{b}_t \in \mathbb{R}^d$ are associated to a unique set of marginal distributions $q(\mathbf{x}_t)$, which are fully characterised by the mean and covariance $\mathbf{M} = (\mathbf{m}_t, \mathbf{S}_t)$. We denote by $\mathbf{V} = (\mathbf{A}_t, \mathbf{b}_t)$ the *variational* parameters of q. The ELBO is given by

$$\mathcal{L}(\mathbf{M}, \mathbf{V}) = \mathbb{E}_{q(\mathbf{x})} \log p(\mathbf{y} \mid \mathbf{x}) + \tfrac{1}{2} \int_0^T \mathbb{E}_{q(\mathbf{x}_t)} \|(\mathbf{A}_t \mathbf{x}_t + \mathbf{b}_t) - f_p(\mathbf{x}_t, t)\|_{\mathbf{Q}_c^{-1}}^2 \mathrm{d}t - \mathrm{D}_{\mathrm{KL}}[q(\mathbf{x}_0) \| p(\mathbf{x}_0)]. \qquad (A41)$$

The constraints connecting the drift parameters to the marginal statistics $(\mathbf{m}, \mathbf{S})$ are

$$C[\mathbf{M}, \mathbf{V}](t) = \begin{bmatrix} \dot{\mathbf{m}}_t - \mathbf{A}_t \, \mathbf{m}_t - \mathbf{b}_t \\ \dot{\mathbf{S}}_t - \mathbf{A}_t \, \mathbf{S}_t - \mathbf{S}_t \, \mathbf{A}_t^\top - \mathbf{Q}_c \end{bmatrix} = \mathbf{0} \qquad \forall t. \qquad (A42)$$

The constrained optimization problem is thus

$$\max_{\mathbf{M}, \mathbf{V}} \mathcal{L}(\mathbf{M}, \mathbf{V}) \qquad (A43)$$

$$\text{subject to } C[\mathbf{M}, \mathbf{V}](t) = \mathbf{0} \qquad \forall t. \qquad (A44)$$

A Lagrangian is constructed, with Lagrangian multipliers $\Theta = (\boldsymbol{\lambda}_t, \boldsymbol{\Psi}_t)$ associated to each of the two constraints as

$$\mathfrak{L}(\mathbf{M}, \mathbf{V}, \Theta) = \mathcal{L}(\mathbf{M}, \mathbf{V}) - \int_0^T \langle \Theta, C[\mathbf{M}, \mathbf{V}](t) \rangle \, \mathrm{d}t. \qquad (A45)$$

A local optimum $(\mathbf{M}^*, \mathbf{V}^*)$ of the original constrained optimization problem is associated to a unique pair of Lagrangian multipliers Θ^* satisfying $\nabla_{\mathbf{M}_t, \mathbf{V}_t, \Theta_t} \mathfrak{L}|_{\mathbf{M}_t^*, \mathbf{V}_t^*, \Theta_t^*} = 0, \forall t$. The system of equations can be re-expressed as:

$$\dot{\boldsymbol{\Psi}}_t^* = -\mathbf{A}_t^{*\top} \boldsymbol{\Psi}_t^* - \boldsymbol{\Psi}_t^* \mathbf{A}_t^* - \nabla_{\mathbf{S}} \mathcal{L}|_{\mathbf{S}^*}, \qquad (A46)$$

$$\dot{\boldsymbol{\lambda}}_t^* = -\mathbf{A}_t^{*\top} \boldsymbol{\lambda}_t^* - \nabla_{\mathbf{m}} \mathcal{L}|_{\mathbf{m}^*}, \qquad (A47)$$

$$\mathbf{A}_t^* = \mathbb{E}_{q^*}[\nabla_{\mathbf{x}} f] - 2\mathbf{Q}_c \boldsymbol{\Psi}^*, \qquad (A48)$$

$$\mathbf{b}_t^* = \mathbb{E}_{q^*}[f] - \mathbf{A}_t^* \mathbf{m}_t^* - \boldsymbol{\lambda}_t^*, \qquad (A49)$$

$$0 = C[\mathbf{m}^*, \mathbf{S}^*, \mathbf{A}^*, \mathbf{b}^*]. \qquad (A50)$$

C.1 Fixed Point Iterations

The system of equations derived earlier are not analytically solvable. They consist of self consistency equations among the optimal variables $\mathbf{x}_t^*, \Theta_t^*$. To find optimal solutions to this problem, a fixed point algorithm, whereby a sequence of variables $(\mathbf{x}_t^{(k)}, \Theta_t^{(k)})$ is constructed in Archambeau et al. (2007a) using the self consistency equations as follows:

From Eq. (A46): $\boldsymbol{\Psi}^{(k+1)} \qquad \leftarrow \dot{\boldsymbol{\Psi}}_t = -\mathbf{A}_t^{(k)\top} \boldsymbol{\Psi}_t - \boldsymbol{\Psi}_t \mathbf{A}_t^{(k)} - \nabla_{\mathbf{S}} \mathcal{L}|_{\mathbf{S}^{(k)}}, \text{ with } \boldsymbol{\Psi}(T) = 0, \qquad (A51)$

From Eq. (A47): $\boldsymbol{\lambda}^{(k+1)} \qquad \leftarrow \dot{\boldsymbol{\lambda}}_t = -\mathbf{A}_t^{(k)\top} \boldsymbol{\lambda}_t - \nabla_{\mathbf{m}} \mathcal{L}|_{\mathbf{m}^{(k)}}, \text{ with } \boldsymbol{\lambda}(T) = 0, \qquad (A52)$

From Eq. (A48): $\mathbf{A}^{(k+1)} \qquad \leftarrow (1-\omega)\mathbf{A}^{(k)} + \omega\left(\mathbb{E}_{q^{(k)}}[\nabla_{\mathbf{x}} f] - 2\mathbf{Q}_c \boldsymbol{\Psi}^{(k+1)}\right), \qquad (A53)$

From Eq. (A49): $\mathbf{b}^{(k+1)} \qquad \leftarrow (1-\omega)\mathbf{b}^{(k)} + \omega\left(\mathbb{E}_{q^{(k)}}[f] - \mathbf{A}_t^{(k+1)}\mathbf{m}_t^{(k)} - \boldsymbol{\lambda}_t^{(k+1)}\right), \qquad (A54)$

From Eq. (A50): $\mathbf{m}^{(k+1)} \qquad \leftarrow \dot{\mathbf{m}}_t - \mathbf{A}_t^{(k+1)} \mathbf{m}_t - \mathbf{b}_t^{(k+1)}, \qquad (A55)$

From Eq. (A50): $\mathbf{S}^{(k+1)} \qquad \leftarrow \dot{\mathbf{S}}_t - \mathbf{A}_t^{(k+1)} \mathbf{S}_t - \mathbf{S}_t \mathbf{A}_t^{(k+1)\top} - \mathbf{Q}_c. \qquad (A56)$

The iterations are run until convergence. In Eqs. (A53) and (A54), the learning-rate ω is introduced for stability reason. It enforces a degree of stickiness to the previous values for $\mathbf{A}$ and $\mathbf{b}$ in the sequence defined by the updates.

D Linear Gaussian Discrete Markov Chains

We consider a Linear Gaussian discrete Markov chain, also known as linear Gaussian state space model (LGSSM), which specifies the distribution of a finite collection of random variables $\mathbf{x} = [\mathbf{x}_0, \ldots, \mathbf{x}_n]$ as follows:

$$\mathbf{x}_0 \sim \mathrm{N}(\mathbf{m}_0, \mathbf{S}_0), \qquad (A57)$$

$$\mathbf{x}_{i+1} = \hat{\mathbf{A}}_i \mathbf{x}_i + \hat{\mathbf{b}}_i + \hat{\varepsilon}_i, \qquad \hat{\varepsilon}_i \sim \mathrm{N}(\mathbf{0}, \hat{\mathbf{Q}}_i). \qquad (A58)$$

The joint density over $\mathbf{x}$ factorises as a chain $p(\mathbf{x}_0, \ldots, \mathbf{x}_n) = p(\mathbf{x}_0) \prod_{i=1}^n p(\mathbf{x}_i \,|\, \mathbf{x}_{i-1})$. It is common in the literature to parametrize the 'drift' via statistics $\boldsymbol{\varphi} = \{\hat{\mathbf{A}}_i, \hat{\mathbf{b}}_i, \hat{\mathbf{Q}}_i\}_{i=1}^n$. It provides an intuitive description of the collection as a temporal process and also an effective way to compute the marginal means $\mathbf{m}_i$, covariances $\mathbf{S}_i$ and cross covariances $\mathbf{C}_i$, defined as

$$\mathbf{m}_{i+1} = \mathbb{E}[\mathbf{x}_{i+1}] = \hat{\mathbf{A}}_i \mathbf{m}_i + \hat{\mathbf{b}}_i, \tag{A59}$$

$$\mathbf{S}_{i+1} = \mathbb{E}[(\mathbf{x}_{i+1} - \mathbf{m}_{i+1})(\mathbf{x}_{i+1} - \mathbf{m}_{i+1})^\top] = \hat{\mathbf{A}}_i \mathbf{S}_i \hat{\mathbf{A}}_i^\top + \hat{\mathbf{Q}}_i, \tag{A60}$$

$$\mathbf{C}_i = \mathbb{E}[(\mathbf{x}_{i+1} - \mathbf{m}_{i+1})(\mathbf{x}_i - \mathbf{m}_i)^\top] = \hat{\mathbf{A}}_i \mathbf{S}_i. \tag{A61}$$

The stacked parameters $\mathbf{M} = \{\mathbf{m}_i, \mathbf{S}_{i+1}, \mathbf{C}_{i+1}\}_{i=1}^n$ constitute a parameterization of the LGSSM. A convenient alternative parameterization of LGSSMs is as a special case of conditional exponential family distributions (Lin et al., 2019)

$$p(\mathbf{x}_{i+1} \,|\, \mathbf{x}_i) = \exp[\langle \mathsf{T}_c(\mathbf{x}_{i+1}, \mathbf{x}_i), \boldsymbol{\lambda}_i \rangle - A_c(\boldsymbol{\lambda}_i)], \tag{A62}$$

$$p(\mathbf{x}_0) = \exp[\langle \mathsf{T}(\mathbf{x}_0), \boldsymbol{\lambda}_0 \rangle - A(\boldsymbol{\lambda}_0)], \tag{A63}$$

where for each conditional distribution, $\mathsf{T}_c(\mathbf{x}_{i+1}, \mathbf{x}_i)$ are the sufficient statistics, $\boldsymbol{\lambda}_i$ the natural parameters, and $A_c(\boldsymbol{\lambda}_i)$ the associated log partition function.

The initial state is distributed as a multivariate Gaussian with sufficient statistics and natural parameters

$$\mathsf{T}(\mathbf{x}_0) = [\mathbf{x}_0, \mathbf{x}_0 \mathbf{x}_0^\top]; \quad \boldsymbol{\lambda}_0 = [\mathbf{S}_0^{-1} \mathbf{m}_0, -\tfrac{1}{2} \mathbf{S}_0^{-1}]. \tag{A64}$$

By expanding the log conditional density as

$$\log p(\mathbf{x}_{i+1} \,|\, \mathbf{x}_i) = -\frac{1}{2}(\mathbf{x}_{i+1} - \hat{\mathbf{A}}_i \mathbf{x}_i - \hat{\mathbf{b}}_i)^\top \hat{\mathbf{Q}}_i^{-1} (\mathbf{x}_{i+1} - \hat{\mathbf{A}}_i \mathbf{x}_i - \hat{\mathbf{b}}_i) + c \tag{A65}$$

$$= \mathbf{x}_{i+1}^\top \hat{\mathbf{Q}}_i^{-1} \hat{\mathbf{b}}_i - \frac{1}{2} \mathbf{x}_{i+1}^\top \hat{\mathbf{Q}}_i^{-1} \mathbf{x}_{i+1} + \mathbf{x}_{i+1}^\top \hat{\mathbf{Q}}_i^{-1} \hat{\mathbf{A}}_i \mathbf{x}_i + \tilde{c}, \tag{A66}$$

we can identify

$$\mathsf{T}(\mathbf{x}_{i+1}, \mathbf{x}_i) = [\mathbf{x}_{i+1}, \mathbf{x}_{i+1} \mathbf{x}_{i+1}^\top, \mathbf{x}_{i+1} \mathbf{x}_i^\top]; \quad \boldsymbol{\lambda}_i = [\hat{\mathbf{Q}}_i^{-1} \hat{\mathbf{b}}_i, -\tfrac{1}{2} \hat{\mathbf{Q}}_i^{-1}, \hat{\mathbf{Q}}_i^{-1} \hat{\mathbf{A}}_i], \tag{A67}$$

where the inner product in Eq. (A63) (resp. Eq. (A62)) distributes over the sufficient statistics in Eq. (A64) (resp. Eq. (A67)) and is the standard inner product $\mathbf{a}, \mathbf{b} \to \mathbf{a}^\top \mathbf{b}$ for vectors and $\mathbf{A}, \mathbf{B} \to \mathrm{Trace}(\mathbf{A}^\top \mathbf{B})$ for matrices.

Finally, the expectation parameters $\boldsymbol{\mu}$ for the conditional exponential family distribution are defined as

$$\boldsymbol{\mu}_i = \mathbb{E}_{p(\mathbf{x}_{i+1}, \mathbf{x}_i)}[\mathsf{T}(\mathbf{x}_{i+1}, \mathbf{x}_i)] = [\mathbf{m}_{i+1}, \ \mathbf{S}_{i+1} + \mathbf{m}_{i+1} \mathbf{m}_{i+1}^\top, \ \mathbf{C}_i + \mathbf{m}_{i+1}], \tag{A68}$$

$$\boldsymbol{\mu}_0 = \mathbb{E}_{p(\mathbf{x}_0)}[\mathsf{T}_0(\mathbf{x}_0)] = [\mathbf{m}_0, \ \mathbf{S}_0 + \mathbf{m}_0 \mathbf{m}_0^\top]. \tag{A69}$$

The joint distribution over states $\mathbf{x}$ is a $(\mathsf{T} + 1)$ d-dimensional multivariate normal distribution in the exponential family

$$p(\mathbf{x}_0, \ldots, \mathbf{x}_N) = \exp[\langle \mathsf{T}(\mathbf{x}), \boldsymbol{\eta}_p \rangle - A(\boldsymbol{\eta}_p)]. \tag{A70}$$

Building the joint density by multiplying the terms from the conditional exponential family description (Eq. (A63) and Eq. (A62)) reveals the sparse structure of the sufficient statistics. Indeed these are the union of the initial and conditional sufficient statistics in Eq. (A64) and Eq. (A67). Importantly, they only include the outer product of identical states $\mathbf{x}_i \mathbf{x}_i^\top$ or consecutive states $\mathbf{x}_i \mathbf{x}_{i+1}^\top$. These sufficient statistics can be conveniently written as $\mathsf{T}(\mathbf{x}) = [\mathbf{x}, \mathrm{btd}(\mathbf{x} \mathbf{x}^\top)]$ where $\mathrm{btd}(\mathbf{M})$ sets entries of $\mathbf{M}$ outside of the d-block tri-diagonals to zero.

An important property we use in the paper is that given two such LGSSMs indexed p_1, p_2 with natural parameters $\boldsymbol{\eta}^{(1)}, \boldsymbol{\eta}^{(2)}$, the Kullback–Leibler divergence is

$$\nabla_{\boldsymbol{\mu}^{(1)}} D_{\mathrm{KL}} \Big[p_1(\mathbf{x}; \boldsymbol{\eta}^{(1)}) \,\|\, p_1(\mathbf{x}; \boldsymbol{\eta}^{(2)}) \Big] = \boldsymbol{\eta}^{(1)} - \boldsymbol{\eta}^{(2)}. \tag{A71}$$

A LGSSM is fully characterized by either of the parameterizations $\boldsymbol{\eta}$, $\boldsymbol{\mu}$, $\boldsymbol{\varphi}$ and $\mathbf{M}$. There is a bijective mapping between each of these parameterizations and CVI-DP requires to compute some of these transformations, which

we derive here for completeness.

φ to **M**:

$$\mathbf{A}_i = \mathbf{C}_i \mathbf{S}_i^{-1}, \tag{A72}$$

$$\mathbf{Q}_i = \mathbf{S}_{i+1} - \mathbf{A}_i \mathbf{S}_i \mathbf{A}_i^\top, \tag{A73}$$

$$\mathbf{b}_i = \mathbf{m}_{i+1} - \mathbf{A}_i \mathbf{m}_i. \tag{A74}$$

M to $\boldsymbol{\mu}$:

$$\boldsymbol{\mu}_i = [\mathbf{m}_{i+1},\ \mathbf{S}_{i+1} + \mathbf{m}_{i+1}\mathbf{m}_{i+1}^\top,\ \mathbf{C}_i + \mathbf{m}_{i+1}\mathbf{m}_i^\top], \tag{A75}$$

$$\boldsymbol{\mu}_0 = [\mathbf{m}_0,\ \mathbf{S}_0 + \mathbf{m}_0\mathbf{m}_0^\top]. \tag{A76}$$

$\boldsymbol{\mu}$ to **M**:

$$[\mathbf{m}_{i+1}, \mathbf{S}_{i+1}, \mathbf{C}_i] = [\boldsymbol{\mu}_i^{(1)},\ \boldsymbol{\mu}_i^{(2)} - \boldsymbol{\mu}_i^{(1)}\boldsymbol{\mu}_i^{(1)\top},\ \boldsymbol{\mu}_i^{(3)} - \boldsymbol{\mu}_i^{(1)}\boldsymbol{\mu}_{i-1}^{(1)\top}], \tag{A77}$$

$$[\mathbf{m}_0,\ \mathbf{S}_0] = [\boldsymbol{\mu}_0^{(1)},\ \boldsymbol{\mu}_0^{(2)} - \mathbf{m}_0\mathbf{m}_0^\top]. \tag{A78}$$

$\boldsymbol{\mu}$ to φ:

$$\mathbf{A}_i = \mathbf{C}_i \mathbf{S}_i^{-1} = (\boldsymbol{\mu}_i^{(3)} - \boldsymbol{\mu}_i^{(1)}\boldsymbol{\mu}_{i-1}^{(1)\top})(\boldsymbol{\mu}_{i-1}^{(2)} - \boldsymbol{\mu}_{i-1}^{(1)}\boldsymbol{\mu}_{i-1}^{(1)\top})^{-1}, \tag{A79}$$

$$\mathbf{b}_i = \mathbf{m}_{i+1} - \mathbf{A}_i \mathbf{m}_i = \boldsymbol{\mu}_i^{(1)} - (\boldsymbol{\mu}_i^{(3)} - \boldsymbol{\mu}_i^{(1)}\boldsymbol{\mu}_{i-1}^{(1)\top})(\boldsymbol{\mu}_{i-1}^{(2)} - \boldsymbol{\mu}_{i-1}^{(1)}\boldsymbol{\mu}_{i-1}^{(1)\top})^{-1}\boldsymbol{\mu}_{i-1}^{(1)}. \tag{A80}$$

E CVI-DP for Linear DP and Arbitrary Observation Model

In this section, we derive the proposed CVI-DP for the model with a linear diffusion process prior and an arbitrary likelihood *i.e.* the prior over the latent state trajectory is a diffusion process, and data $\mathcal{D} = \{(t_i, y_i)\}_{i=1}^n$ consist of noisy observations of the process, $y_i = \mathbf{h}^\top \mathbf{x}_{t_i} + \epsilon_i$ with *i.i.d.* noise $\{\epsilon_i\}_{i=1}^n$. Therefore, the likelihood function factorizes as $p(\mathbf{y} \mid \mathbf{x}) = \prod_{i=1}^n p(y_i \mid \mathbf{h}^\top \mathbf{x}_{t_i})$. Note that, although we derive the method for the specific *i.i.d.* setting with a linear projection from states to observations, the algorithm is not restricted to these assumptions.

E.1 Inference

When the observation model is not Gaussian, we can still marginalize the diffusion to a finite Markov chain. However, the posterior no longer belongs to an exponential family $\mathcal{F}$, and we need to resort to approximate inference. Under the variational framework, restricting q to belong to $\mathcal{F}$ has the convenient property that the optimal posterior has the same additive decomposition as in the conjugate case: $\boldsymbol{\eta}_q^* = \boldsymbol{\eta}_p + \phi(\boldsymbol{\lambda}^*)$, with $\boldsymbol{\lambda}^* = (\boldsymbol{\lambda}_1^*, \boldsymbol{\lambda}_2^*)$. This is revealed by looking at the first-order stationary condition of the optimal distribution q^*. The distributions in $\mathcal{F}$ can be parameterized via their natural parameters $\boldsymbol{\eta}$ but also equivalently by their expectation parameters $\boldsymbol{\mu} = \mathbb{E}[\mathsf{T}(\mathbf{x})]$. The gradient of the ELBO with respect to the expectation parameters $\boldsymbol{\mu}$ of q is given by $\nabla_{\boldsymbol{\mu}}\mathcal{L} = \nabla_{\boldsymbol{\mu}}\mathbb{E}_q[\log p(\mathbf{y} \mid \mathbf{x})] - (\boldsymbol{\eta}_q - \boldsymbol{\eta}_p)$, where we used the property $\nabla_{\boldsymbol{\mu}}\mathrm{D}_{\mathrm{KL}}[q \,\|\, p] = \boldsymbol{\eta}_q - \boldsymbol{\eta}_p$. At the optimum,

$$\nabla_{\boldsymbol{\mu}}\mathcal{L}|_{\boldsymbol{\mu}^*} = \mathbf{0} \implies \boldsymbol{\eta}_q^* = \boldsymbol{\eta}_p + \nabla_{\boldsymbol{\mu}}\mathbb{E}_q[\log p(y_i \mid \mathbf{x})]|_{\boldsymbol{\mu}^*}. \tag{A81}$$

Given the factorizing likelihood, noting $u_i = \mathbf{h}^\top \mathbf{x}_{t_i}$,

$$\mathbb{E}_{q(\mathbf{x})}[\log p(\mathbf{y} \mid \mathbf{x})]|_{\boldsymbol{\mu}^*} = \sum_i \mathbb{E}_{q(u_i)}[\log p(y_i \mid \mathbf{h}^\top \mathbf{x}_{t_i} = u_i)]|_{\boldsymbol{\mu}_i^*}, \tag{A82}$$

where $q(u_i) = \mathrm{N}(m_i{=}\mathbf{h}^\top\mathbf{m}_i, v_i{=}\mathbf{h}^\top\mathbf{S}_i\mathbf{h})$ and $\boldsymbol{\mu}_i = [\mathbf{m}_i, \mathbf{S}_i + \mathbf{m}_i\mathbf{m}_i^\top]$. We introduce the scalar gradients as

$$\alpha_i = \nabla_{m_i}\mathbb{E}_{q(u_i)}[g_i(u_i)]|_{m_i^*}, \tag{A83}$$

$$\beta_i = \nabla_{v_i}\mathbb{E}_{q(u_i)}[g_i(u_i)]|_{v_i^*}, \tag{A84}$$

where $g_i(u_i) = \log p(y_i \mid u_i)$. The chain rule gives

$$\nabla_{\boldsymbol{\mu}_i}\mathbb{E}_{q(u_i)}[g(u_i)]|_{\boldsymbol{\mu}_i^*} = (\mathbf{h}(\alpha_i^* - 2\beta_i^*\boldsymbol{\mu}_i^*), \mathbf{h}\mathbf{h}^\top\beta_i^*) \triangleq (\mathbf{h}\lambda_{i,1}^*, \mathbf{h}\mathbf{h}^\top\lambda_{i,2}^*), \tag{A85}$$

which reveals the low rank structure of these gradients. Stacking the $(\lambda_{i,1}^*, \lambda_{i,2}^*)$ into the matrix $\boldsymbol{\lambda} \in \mathbb{R}^{n \times 2}$, we introduce the bijective linear operator ϕ with span $\mathcal{S} \subset \mathbb{R}^{nd} \times \mathbb{R}^{nd \times nd}$,

$$\phi : \mathbb{R}^{n \times 2} \to \mathcal{S}, \quad \langle \mathsf{T}(\mathbf{x}), \phi(\boldsymbol{\lambda}) \rangle = \sum_i \langle \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i), \boldsymbol{\lambda}_i \rangle. \tag{A86}$$

Thus, the optimality condition in Eq. (A81) can be rewritten as $\boldsymbol{\eta}_q^* = \boldsymbol{\eta}_p + \phi(\boldsymbol{\lambda}^*)$. Therefore, to find the optimal variational parameters, it is sufficient to search the space of distribution in $\mathcal{F}$ with natural parameters $\boldsymbol{\eta}_q = \boldsymbol{\eta}_p + \phi(\boldsymbol{\lambda})$.

Conjugate-computation VI (CVI, Khan and Lin, 2017) gives an efficient algorithm to find the optimal $\boldsymbol{\lambda}^*$ by running mirror descent with the Kullback–Leibler divergence (KL) as Bregman divergence. More precisely, using ρ step-size, it constructs a sequence of iterates $\boldsymbol{\lambda}^{(k)}$ via

$$\boldsymbol{\mu}^{(k+1)} = \arg\max_{\boldsymbol{\mu}} \langle \nabla_{\boldsymbol{\mu}} \mathcal{L}(\boldsymbol{\mu}^{(k)}), \boldsymbol{\mu} \rangle - \frac{1}{\rho} \mathrm{D}_{\mathrm{KL}} \Big[q(\mathbf{x}; \boldsymbol{\mu}) \,\|\, q(\mathbf{x}; \boldsymbol{\mu}^{(k)}) \Big]. \tag{A87}$$

This maximization can be computed in closed-form leading to the following updates in the natural parameterization

$$\boldsymbol{\lambda}^{(k+1)} = (1 - \rho)\, \boldsymbol{\lambda}^{(k)} + \rho\, \phi^{-1} \big(\nabla_{\boldsymbol{\mu}} \mathbb{E}_{q^{(k)}} \log p(\mathbf{y} \,|\, \mathbf{x}) \big). \tag{A88}$$

This global update rule can be decomposed into n local updates as

$$\boldsymbol{\lambda}_i^{(k+1)} = (1 - \rho)\, \boldsymbol{\lambda}_i^{(k)} + \rho\, \phi_i^{-1} \big(\nabla_{\boldsymbol{\mu}_i} \mathbb{E}_{q^{(k)}} \log p(y_i \,|\, \mathbf{x}_i) \big), \tag{A89}$$

where we introduced the bijective linear operator ϕ_i with span $\mathcal{S}_i \subset \mathbb{R}^d \times \mathbb{R}^{d \times d}$,

$$\phi_i : \mathbb{R}^2 \to \mathcal{S}_i, \quad \langle \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i), \boldsymbol{\lambda}_i \rangle = \langle \mathsf{T}(\mathbf{x}), \phi_i(\boldsymbol{\lambda}_i) \rangle. \tag{A90}$$

In the case of Gaussian observations, the gradient term in the updates is independent of where the gradient is evaluated and equal to $\boldsymbol{\lambda}^*$, so a *single step* of CVI with step-size $\rho = 1$ leads to the optimum (see Fig. A8 for an empirical example). In the more general setting of log-concave likelihoods, the iterations in Eq. (A88) are guaranteed to converge to the optimum.

E.2   Learning

Standard VEM procedures iterate pairs of E (Expectaction) and M (Maximization) steps to built a sequence of variational distributions $\{q_t\}_{t=1}^K$ and parameters $\{\boldsymbol{\theta}_t\}_{t=1}^K$ which we here describe in the context of the dual parameterization. In the E-step, starting from a hyperparameter $\boldsymbol{\theta}_t$, the optimal variational distribution $q_t = \arg\max_{q \in \mathcal{Q}} \mathcal{L}(q, \boldsymbol{\theta}_t)$ is computed via maximizing the ELBO. For the dual parameterization, we get the optimal variational parameters as $\boldsymbol{\eta}_{q_t}^* = \boldsymbol{\eta}_p(\boldsymbol{\theta}_t) + \boldsymbol{\lambda}_t^*$. The standard M-step then finds $\boldsymbol{\theta}_{t+1} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(q_t, \boldsymbol{\theta})$. Noting $\mathcal{L}_\eta(\boldsymbol{\eta}_{q_t}, \boldsymbol{\theta}) = \mathcal{L}(q_t, \boldsymbol{\theta})$, we can rewrite the M-step as

$$\text{Standard M-step:} \quad \boldsymbol{\theta}_{t+1} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_\eta(\boldsymbol{\eta}_p(\boldsymbol{\theta}_t) + \boldsymbol{\lambda}_t^*, \boldsymbol{\theta}). \tag{A91}$$

Instead of performing coordinate ascent on $\mathcal{L}(q_t, \boldsymbol{\theta})$, we do so on $l(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \mathcal{L}_\eta(\boldsymbol{\eta}_p(\boldsymbol{\theta}) + \boldsymbol{\lambda}, \boldsymbol{\theta})$, which preserves the original E-step but leads to the modified M-step

$$\text{Proposed M-step:} \quad \boldsymbol{\theta}_{t+1} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_\eta(\boldsymbol{\eta}_p(\boldsymbol{\theta}) + \boldsymbol{\lambda}_t^*, \boldsymbol{\theta}). \tag{A92}$$

This proposed objective is still a lower bound to the marginal likelihood and it corresponds to the ELBO in Eq. (4) with a distribution whose natural parameter is $\hat{\boldsymbol{\eta}}_{q_t}(\boldsymbol{\theta}) = \boldsymbol{\eta}_p(\boldsymbol{\theta}) + \boldsymbol{\lambda}_t^*$ and thus depends on $\boldsymbol{\theta}$. The ELBO is different from the distribution obtained after the E-step, with natural parameter $\boldsymbol{\eta}_{q_t}^*$ which is independent of $\boldsymbol{\theta}$. We denote this distribution by $q(\mathbf{x}; \boldsymbol{\eta}_t^*)$. Clearly, at $\boldsymbol{\theta} = \boldsymbol{\theta}_t$ both objectives match, and so do their gradient with respect to $\boldsymbol{\theta}$, but they generally differ otherwise. Adam et al. (2021) showed that this parameterization leads to a tighter lower bound. Therefore, the learning objective can be expressed as

$$\mathcal{L}(\underbrace{\boldsymbol{\eta}_p(\boldsymbol{\theta}) + \lambda^*}_{\boldsymbol{\eta}_q(\boldsymbol{\theta})}, \boldsymbol{\theta}) = \mathbb{E}_{q_{\boldsymbol{\eta}(\boldsymbol{\theta})}(\mathbf{x})} \left[\log \frac{\prod_{i=1}^n p(y_i \,|\, \mathbf{x}_i) p_{\boldsymbol{\theta}}(\mathbf{x})}{\frac{1}{\mathcal{Z}(\boldsymbol{\theta})} \prod_{i=1}^n t_i^*(\mathbf{x}_i) p_{\boldsymbol{\theta}}(\mathbf{x})} \right]$$

$$= \log \mathcal{Z}_t(\boldsymbol{\theta}) + \underbrace{\sum_{i=1}^n \mathbb{E}_{q_{\boldsymbol{\eta}(\boldsymbol{\theta})}(\mathbf{x})} \left[\log \frac{p(y_i \,|\, \mathbf{x}_i)}{t_i^*(\mathbf{x}_i)} \right]}_{c(\boldsymbol{\theta})}, \tag{A93}$$

where $\log \mathcal{Z}_t(\boldsymbol{\theta})$ is the log-partition of $\prod_{i=1}^{n} t_i^*(\mathbf{x}_i) p_{\boldsymbol{\theta}}(\mathbf{x})$. It is the log marginal likelihood of a generative model where the prior is the linear DP, $p_{\boldsymbol{\theta}}(\mathbf{x})$, and the observation likelihoods are given by the Gaussian sites. Hence, this is classic Gaussian process regression. The term $c(\boldsymbol{\theta})$ captures a notion of mismatch between the true likelihood terms and their Gaussian approximation via sites which is $\mathbf{0}$ in the setting of Gaussian observations and matched sites.

F CVI-DP for Non-linear DPs (Discrete Case)

We discretize the prior process on an ordered time-grid $\tau = (t_1, \dots, t_M) \in [0, T]$ using Euler–Maruyama as

$$p(\{\mathbf{x}\}_\tau) \prod_{m=0}^{M-1} \mathrm{N}(\mathbf{x}_m + f_p(\mathbf{x}_m)\Delta t, \mathbf{Q}_c \Delta t), \tag{A94}$$

where we denote the discretized process as $\{\mathbf{x}\}_\tau$. We assume the discretized process is observed at $n < M$ time points and denote by $\mathbf{x}^d$ the vector of states such that $p(y_i \,|\, \{\mathbf{x}\}_\tau) = p(y_i \,|\, \mathbf{x}_i^d)$. The CVI algorithm applies to inference problems with exponential family priors. We denote by $\mathcal{F}_c$ the set of Markovian Gaussian processes marginalized to $\{\mathbf{x}\}_\tau$, which is a Gaussian exponential family with sufficient statistics $\mathsf{T}(\{\mathbf{x}\}_\tau)$.

F.1 Details of Change of Measure

We introduce $p_L \in \mathcal{F}_c$ with linear drift $f_L(\mathbf{x}_m, m) = \mathbf{A}_m \mathbf{x}_m + \mathbf{b}_m$ and the added constraint that it shares the same innovation noise as the prior p. Thus, the change of measure is

$$\frac{p(\{\mathbf{x}\}_\tau)}{p_L(\{\mathbf{x}\}_\tau)} = \exp\Big(\sum_{m=0}^{M-1} \log \frac{p(\mathbf{x}_{m+1} \,|\, \mathbf{x}_m)}{p_L(\mathbf{x}_{m+1} \,|\, \mathbf{x}_m)}\Big) \tag{A95}$$

$$= \exp\Big(\sum_{m=0}^{M-1} V(\tilde{\mathbf{x}}_m)\,\Delta t\Big), \tag{A96}$$

where we write the individual sum terms as $V(\tilde{\mathbf{x}}_m)\,\Delta t$ and $\tilde{\mathbf{x}}_m = [\mathbf{x}_{m+1}, \mathbf{x}_m]$ to denote consecutive pairs of states. We can further expand it as

$$V(\tilde{\mathbf{x}}_m) = \frac{1}{\Delta t} \log \frac{p(\mathbf{x}_{m+1} \,|\, \mathbf{x}_m)}{p_L(\mathbf{x}_{m+1} \,|\, \mathbf{x}_m)} \tag{A97}$$

$$= \frac{1}{2}\Big\|\frac{\mathbf{x}_{m+1} - \mathbf{x}_m}{\Delta t} - f_p(\mathbf{x}_m)\Big\|_{\mathbf{Q}_c^{-1}}^2 - \frac{1}{2}\Big\|\frac{\mathbf{x}_{m+1} - \mathbf{x}_m}{\Delta t} - f_L(\mathbf{x}_m)\Big\|_{\mathbf{Q}_c^{-1}}^2 \tag{A98}$$

$$= \frac{1}{2}\Big\|\frac{\mathbf{x}_{m+1} - \mathbf{x}_m}{\Delta t} - f_L(\mathbf{x}_m) + f_L(\mathbf{x}_m) - f_p(\mathbf{x}_m)\Big\|_{\mathbf{Q}_c^{-1}}^2 - \frac{1}{2}\Big\|\frac{\mathbf{x}_{m+1} - \mathbf{x}_m}{\Delta t} - f_L(\mathbf{x}_m)\Big\|_{\mathbf{Q}_c^{-1}}^2 \tag{A99}$$

$$= \frac{1}{2}\|f_L(\mathbf{x}_m) - f_p(\mathbf{x}_m)\|_{\mathbf{Q}_c^{-1}}^2 + \Delta t \Big\langle \frac{\mathbf{x}_{m+1} - \mathbf{x}_m}{\Delta t} - f_L(\mathbf{x}_m),\, f_p(\mathbf{x}_m) - f_L(\mathbf{x}_m) \Big\rangle_{\mathbf{Q}_c^{-1}}, \tag{A100}$$

where the time dependence of the drift function is dropped to simplify the notation. Thus, we obtain an alternative expression for the posterior distribution

$$p(\{\mathbf{x}\}_\tau \,|\, \mathcal{D}) = p_L(\{\mathbf{x}\}_\tau)\, p(\mathbf{y} \,|\, \{\mathbf{x}\}_\tau) \exp\Big(\sum_{m=0}^{M-1} V(\tilde{\mathbf{x}}_m)\,\Delta t\Big). \tag{A101}$$

This corresponds to an inference problem with the prior over the latent chain having density p_L, and two likelihoods: the original and now *sparse* likelihood $p(\mathbf{y} \,|\, \{\mathbf{x}\}_\tau)$ and an additional *dense* likelihood stemming from the change of measure.

F.2 Solving the Discretized Problem

In the discrete form, the intractable posterior can be approximated by introducing a Markovian Gaussian process parameterized via a mixture of dense and sparse sites as done in Cseke et al. (2013, 2016):

$$q(\{\mathbf{x}\}_\tau) \propto p_L(\{\mathbf{x}\}_\tau) \exp\Big(\sum_{i=1}^{n} \langle \boldsymbol{\lambda}_i, \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i^d) \rangle + \sum_{m=0}^{M-1} \langle \boldsymbol{\Lambda}_m, \mathsf{T}(\tilde{\mathbf{x}}_m) \rangle \Delta t \Big). \tag{A102}$$

The approximating distribution q belongs to $\mathcal{F}_c$ and has natural parameters $\boldsymbol{\eta}_q = \boldsymbol{\eta}_L + \phi(\boldsymbol{\lambda}) + \psi(\boldsymbol{\Lambda})$, where ϕ and ψ are linear operators (see App. F.3).

F.3 Structure of the Optimal Posterior

In the discretized CVI-DP method, the ELBO is

$$\mathcal{L}(q) = -\mathrm{D}_{\mathrm{KL}}[q(\{\mathbf{x}\}_\tau \,\|\, p_L(\{\mathbf{x}\}_\tau)] + \mathbb{E}_{q(\mathbf{x})} \log p(\mathbf{y}\,|\,\mathbf{x}) + \sum_{m=0}^{M-1} \mathbb{E}_{q(\tilde{\mathbf{x}}_m)} V(\tilde{\mathbf{x}}_m)\,\Delta t\,.$$

The optimality condition in this scenario is obtained as

$$\nabla_{\boldsymbol{\mu}}\mathcal{L}|_{\boldsymbol{\mu}^*} = \mathbf{0} \implies \boldsymbol{\eta}_q^* = \boldsymbol{\eta}_L + \nabla_{\boldsymbol{\mu}}\mathbb{E}_{q(\mathbf{x})} \log p(\mathbf{y}\,|\,\mathbf{x}) + \sum_{m=0}^{M-1} \nabla_{\boldsymbol{\mu}}\mathbb{E}_{q(\tilde{\mathbf{x}}_m)} V(\tilde{\mathbf{x}}_m)\,\Delta t. \tag{A103}$$

Introducing the bijective linear mappings ψ_m with span $\mathcal{S}_m \subset \mathbb{R}^{Md} \times \mathbb{R}^{Md \times Md}$,

$$\psi_m : \mathbb{R}^{2d} \times \mathbb{R}^{2d \times 2d} \to \mathcal{S}_m : \langle \boldsymbol{\Lambda}_m, \mathsf{T}(\tilde{\mathbf{x}}_m) \rangle = \langle \mathsf{T}(\mathbf{x}), \psi_m(\boldsymbol{\Lambda}_m) \rangle\,, \tag{A104}$$

and the bijective linear operator ϕ with span $\mathcal{S} \subset \mathbb{R}^{nd} \times \mathbb{R}^{nd \times nd}$,

$$\phi : \mathbb{R}^{n \times 2} \to \mathcal{S} : \langle \mathsf{T}(\mathbf{x}), \phi(\boldsymbol{\lambda}) \rangle = \sum_i \langle \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i), \boldsymbol{\lambda}_i \rangle. \tag{A105}$$

There exists $\boldsymbol{\lambda}^* \in \mathbb{R}^{n \times 2}$ and $\boldsymbol{\Lambda}^* \in \mathbb{R}^{M \times 2d} \times \mathbb{R}^{M \times 2d \times 2d}$

$$\boldsymbol{\eta}_q^* = \boldsymbol{\eta}_L + \phi(\boldsymbol{\lambda}^*) + \sum_{m=0}^{M-1} \psi_m(\boldsymbol{\Lambda}_m^*)\,\Delta t. \tag{A106}$$

F.4 Derivation of the CVI-DP Update Rule

The inference algorithm consists of applying Mirror descent on the ELBO in the expectation parameterization,

$$\boldsymbol{\mu}^{(k+1)} = \arg\max_{\boldsymbol{\mu}} \langle \nabla_{\boldsymbol{\mu}}\mathcal{L}(\boldsymbol{\mu}^{(k)}), \boldsymbol{\mu} \rangle - \rho^{-1}\mathrm{D}_{\mathrm{KL}}\left[q(\mathbf{x};\boldsymbol{\mu}) \,\|\, q(\mathbf{x};\boldsymbol{\mu}^{(k)})\right]. \tag{A107}$$

This leads to the following *local* updates

$$\boldsymbol{\lambda}_i^{(k+1)} = (1-\rho)\boldsymbol{\lambda}_i^{(k)} + \rho\phi_i^{-1}\left(\nabla_{\boldsymbol{\mu}_i}\mathbb{E}_{q^{(k)}} \log p(y_i\,|\,\mathbf{x}_i^d)\right), \tag{A108}$$

$$\boldsymbol{\Lambda}_m^{(k+1)} = (1-\rho)\boldsymbol{\Lambda}_m^{(k)} + \rho\psi_m^{-1}\left(\nabla_{\tilde{\boldsymbol{\mu}}_m}\mathbb{E}_{q^{(k)}} V(\tilde{\mathbf{x}}_m)\right). \tag{A109}$$

F.5 Learning

We adapted the modified VEM procedure described in App. E.2 to the non-linear DP setting.

In the non-linear DP case, the objective for the modified M-Step can be expressed as:

$$\mathcal{L}(\underbrace{\boldsymbol{\eta}_{p_L}(\boldsymbol{\theta}) + \Lambda^*}_{\boldsymbol{\eta}_q(\boldsymbol{\theta})}, \boldsymbol{\theta}) = \mathbb{E}_{q_{\boldsymbol{\eta}(\boldsymbol{\theta})}(\mathbf{x})}\left[\log \frac{\prod_{i=1}^n p(y_i\,|\,\mathbf{x}_i)p_{\boldsymbol{\theta}}(\mathbf{x})}{\frac{1}{\mathcal{Z}(\boldsymbol{\theta})}\prod_{i=1}^n t_i^*(\mathbf{x}_i)p_{L\boldsymbol{\theta}}(\mathbf{x})}\right]$$

$$= \log \mathcal{Z}_t(\boldsymbol{\theta}) + \underbrace{\mathbb{E}_{q_{\boldsymbol{\eta}(\boldsymbol{\theta})}(\mathbf{x})}\left[\log \frac{p_{\boldsymbol{\theta}}(\mathbf{x})}{p_{L\boldsymbol{\theta}}(\mathbf{x})}\right]}_{w(\boldsymbol{\theta})} + \underbrace{\sum_{i=1}^n \mathbb{E}_{q_{\boldsymbol{\eta}(\boldsymbol{\theta})}(\mathbf{x})}\left[\log \frac{p(y_i\,|\,\mathbf{x}_i)}{t_i^*(\mathbf{x}_i)}\right]}_{c(\boldsymbol{\theta})}, \tag{A110}$$

where $\log \mathcal{Z}_t(\boldsymbol{\theta})$ is the log-partition of $\prod_{i=1}^n t_i^*(\mathbf{x}_i)p_{L\boldsymbol{\theta}}(\mathbf{x})$. It is the log marginal likelihood of a generative model where the prior is the linear DP, $p_{L\theta}(\mathbf{x})$, and the observation likelihoods are given by the Gaussian sites. The term $c(\boldsymbol{\theta})$ captures a notion of mismatch between the true likelihood terms and their Gaussian approximation via sites which is $\mathbf{0}$ in the setting of a Gaussian observation model. The term $w(\boldsymbol{\theta})$ is related to the error introduced by the linearization of the non-linear drift of the prior DP. This error is $\mathbf{0}$ when the prior is a linear DP.

G CVI-DP for Non-linear DPs (Continuous Case)

In this section, we describe the procedure of taking the size of the time grid to infinity. We keep the time horizon T fixed, but increase the number M of time points. Taking $M \to \infty$ is equivalent to taking $\Delta t = \frac{T}{M-1} \to 0$.

Algorithm 2: CVI-DP inference under a non-linear DP prior (continuous version).

Input: Prior p, q, data $\mathcal{D}$, learning rate ρ
while *not converged* **do**
    while *not converged* **do**
        Sites update:
        $\boldsymbol{\lambda}_i^{(k+1)} = (1-\rho)\boldsymbol{\lambda}_i^{(k)} + \rho\,\phi_i^{-1}(\nabla_{\boldsymbol{\mu}_i}\mathbb{E}_{q^{(k)}}\log p(y_i\,|\,\mathbf{x}_i^{\mathrm{d}}))$
        $\boldsymbol{\Lambda}_t^{(k+1)} = (1-\rho)\boldsymbol{\Lambda}_t^{(k)} + \rho\,\psi_t^{-1}(\nabla_{\boldsymbol{\mu}_t}\mathbb{E}_{q^{(k)}}V(\mathbf{x}_t))$
        Compute posterior:
        $\boldsymbol{\eta}_q^{(k+1)} = \boldsymbol{\eta}_{p_L} + \phi(\boldsymbol{\lambda}^{(k+1)}) + \psi(\boldsymbol{\Lambda}^{(k+1)})$
    end
    Update p_L via posterior linearization:
    $(\mathbf{A}_t, \mathbf{b}_t)^{new} \leftarrow \Pi_{q(\mathbf{x}_t)}[f_p(\cdot, t)]$
    Update $\boldsymbol{\Lambda}$ under p_L^{new}: $\boldsymbol{\Lambda}|_{p_L^{\mathrm{new}}} = \boldsymbol{\Lambda}|_{p_L} + \boldsymbol{\eta}_{p_L} - \boldsymbol{\eta}_{p_L^{\mathrm{new}}}$
end

First, we look at the change of measure in Eq. (A100). For $t \in [0, T]$, we consider $m(M) = \frac{tM}{T}$ and take $M \to \infty$,

$$\lim_{M \to \infty} V(\tilde{\mathbf{x}}_{m(M)}) = V(\mathbf{x}_t) = \frac{1}{2}\|f_L(\mathbf{x}_t) - f_p(\mathbf{x}_t)\|_{\mathbf{Q}_c^{-1}}^2. \tag{A111}$$

As we take the dense grid limit, the change of measure is instantaneous, *i.e.*, it no longer depends on pairs of consecutive states on the grid.

If we take a similar limit to the update rule Eq. (A109), we have that $\boldsymbol{\Lambda}_t \in \mathbb{R}^d \times \mathbb{R}^{d \times d}$, and,

$$\boldsymbol{\Lambda}_t^{(k+1)} = (1-\rho)\boldsymbol{\Lambda}_t^{(k)} + \rho\left(\nabla_{\boldsymbol{\mu}_t}\mathbb{E}_{q^{(k)}}V(\mathbf{x}_t)\right). \tag{A112}$$

We believe these updates could be obtained by introducing the continuous version of the exponential family for Gaussian Markov chain in Eq. (A70) and using functional Mirror descent in place of the standard mirror descent we used in Eq. (A107).

The limit of the variational posterior in Eq. (A102) is a Markovian Gaussian process and no longer has a density with respect to the Lebesgue measure. Informally, we write

$$q(\mathbf{x}) \propto p_L(\mathbf{x})\exp\left(\sum_{i=1}^n \langle \boldsymbol{\lambda}_i, \mathsf{T}(\mathbf{h}^\top \mathbf{x}_i^d)\rangle + \int_0^T \langle \boldsymbol{\Lambda}_t, \mathsf{T}(\mathbf{x}_t)\rangle \,\mathrm{d}t\right). \tag{A113}$$

Note that the sufficient statistics for each dense site Λ_t is now $\mathsf{T}(\mathbf{x}_t) \in \mathbb{R}^d \times \mathbb{R}^{d \times d}$ instead of $\mathsf{T}(\tilde{\mathbf{x}}_m) \in \mathbb{R}^{2d} \times \mathbb{R}^{2d \times 2d}$. In practice, this means that after taking the continuous time limit, the variational process q shares the same diffusion coefficient as the base process p_L, hence of the prior p.

To evaluate the updates in Eq. (A112), we need to compute the marginal statistics (means and covariances) of the variational process at all times. These are available via Kalman-Bucy smoothing (see Appendix B.1, Cseke et al., 2013).

The full continuous time algorithm is in Alg. 2. After taking the limit, the resulting algorithm no longer depends on the Euler–Maruyama discretization scheme we introduced to build the discretized inference algorithm. Just as for the algorithm of Archambeau et al. (2007a), many discretization schemes may be chosen to implement Alg. 2.

H   Monte Carlo Baselines

In the main paper, we compare to baseline 'ground-truth' solutions both for inferring the latent processes and for parameter learning targets. For inference, the problem falls under sequential Monte Carlo (particle smoothing) methods, and for estimating the approximate marginal likelihood we employ annealed importance sampling (AIS). In practice, we could use any simulation methods for the baseline, but we settle for the following setup due to its robustness and fast execution.
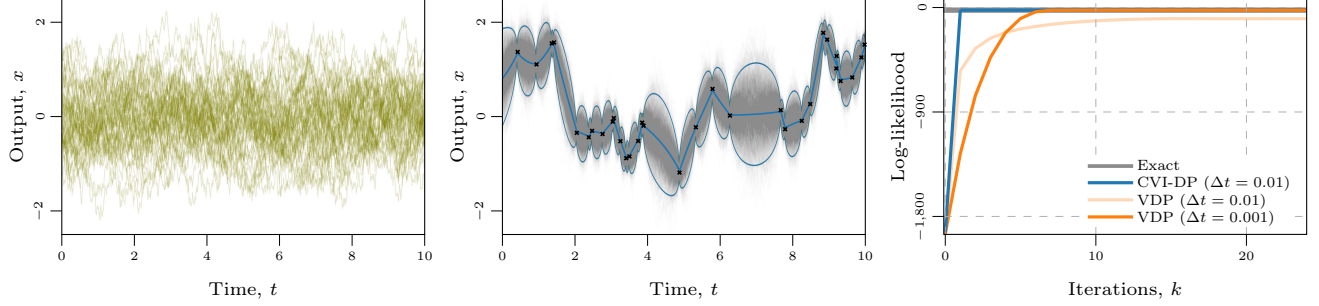
Figure A8: Approximate inference under a linear diffusion process (Ornstein–Uhlenbeck). Left: Draws from the prior. Middle: Approximate posterior process for CVI-DP overlaid over SMC ground-truth samples. Right: CVI-DP converges quickly even with large discretization step when inferring the variational parameters, while VDP suffers from slow convergence even with a small discretization step.

H.1 Sequential Monte Carlo (SMC) Baseline

We use a sequential Monte Carlo approach in the form of particle smoothing through conditional particle filtering with ancestor sampling. The posterior (smoothing) distribution $p(\mathbf{x} \mid \mathbf{y})$ problem can not be computed in closed form in the general case due to the nonlinear nature of the models. We employ the approach of Svensson et al. (2015), where the smoothing distribution $p(\mathbf{x} \mid \mathbf{y})$ is approximated by generating K (correlated) samples using sequential Monte Carlo. Each iteration of the MCMC algorithm uses a conditional particle filter with ancestor sampling. This approach has been shown to avoid the problem of particle degeneracy typically occurring in particle filters (Svensson et al., 2015).

As our setting is continuous-discrete (continuous-time prior with observations discretely spread over the time-horizon), we choose to use an Euler–Maruyama scheme with time-step size Δt for solving the SDE prior for discrete-time steps.

H.2 Annealed Importance Sampling (AIS)

As an illustrative ground-truth for the parameter learning target (marginal likelihood), we use a sampling approach as the baseline. We use annealed importance sampling (AIS, Neal, 2001) which is similar to that used for Gaussian processes with non-conjugate likelihoods in Kuss and Rasmussen (2005) and Nickisch and Rasmussen (2008). The setup defines a sequence of $j = 0, 1, \ldots, J$ steps $Z_j = \int p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j)} p(\mathbf{x}; \theta) \, \mathrm{d}\mathbf{x}$, where $\tau(j) = (j/J)^4$ (such that $\tau(0) = 0$ and $\tau(J) = 1$). The marginal likelihood can be rewritten as

$$p(\mathbf{y}; \theta) = \frac{Z_J}{Z_0} = \frac{Z_J}{Z_{J-1}} \frac{Z_{J-1}}{Z_{J-2}} \cdots \frac{Z_1}{Z_0}, \tag{A114}$$

where Z_j/Z_{j-1} is approximated by importance sampling using samples from $q_j(\mathbf{x}) \propto p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j-1)} p(\mathbf{x}; \theta)$:

$$\frac{Z_j}{Z_{j-1}} = \frac{\int p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j)} p(\mathbf{x}; \theta) \, \mathrm{d}\mathbf{x}}{Z_{j-1}} \tag{A115}$$

$$= \int \frac{p(\mathbf{y} \mid \mathbf{f}; \theta)^{\tau(j)}}{p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j-1)}} \frac{p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j-1)} p(\mathbf{x}; \theta)}{Z_{t-1}} \, \mathrm{d}\mathbf{x} \tag{A116}$$

$$\approx \frac{1}{S} \sum_{s=1}^{S} p(\mathbf{y} \mid \mathbf{x}_j^{(s)}; \theta)^{\tau(j) - \tau(j-1)}, \quad \text{where} \quad \mathbf{x}_j^{(s)} \sim \frac{p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j-1)} p(\mathbf{x}; \theta)}{Z_{j-1}}. \tag{A117}$$

The difference to Kuss and Rasmussen (2005) and Nickisch and Rasmussen (2008) is that here sampling $\mathbf{x}_j$ from $p(\mathbf{y} \mid \mathbf{x}; \theta)^{\tau(j-1)} p(\mathbf{x}; \theta)/Z_{j-1}$ is non-trivial due to the non-linear/non-Gaussian nature of the prior. Here we use sequential Monte Carlo in the form of particle smoothing through conditional particle filtering with ancestor sampling (i.e., the method presented in App. H.1) to draw those samples. For each process $\mathbf{x}_j$ sampled from the SMC approach we only use the 10th sample (discarding the preceding ones as burn-in) and otherwise follow the setup described in App. H.1.
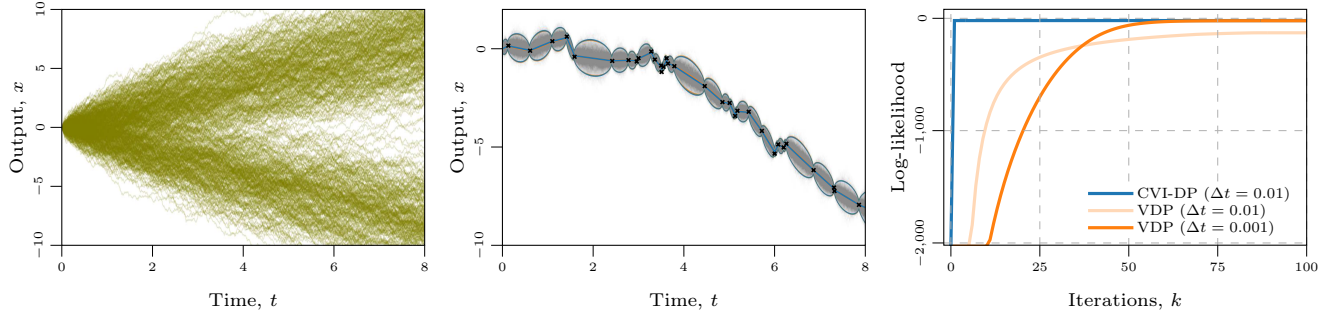
Figure A9: Approximate inference under a Beneš prior (draws from prior on the left). Middle: Approximate posterior processes for CVI-DP and VDP overlaid on the SMC ground-truth samples. Right: CVI-DP converges quickly even with large discretization step when inferring the variational parameters, while VDP suffers from slow convergence even with a small discretization step.

By using a single sample $S = 1$ and a large number of steps J, the estimation of the log marginal likelihood can be written as

$$\log p(\mathbf{y}; \theta) = \sum_{j=1}^{J} \log \frac{Z_j}{Z_{j-1}} \approx \sum_{j=1}^{J} (\tau(j) - \tau(j-1)) \log p(\mathbf{y} \mid \mathbf{x}_j; \theta). \tag{A118}$$

Following Kuss and Rasmussen (2005), we set $J = 8000$ and combine three estimates of log marginal likelihood by their geometric mean.

I   Details on Experiments

In this section, we provide details for the experiments which were included in Sec. 5. We start by explaining the setup and details about the synthetic tasks for each diffusion process individually in App. I.1. We then cover the details about the experiments on real-world finance data and vehicle tracking in App. I.2 and App. I.3. We perform a grid search in all the experiments to find the best learning rate and other hyperparameters for all the methods.

I.1   Synthetic Tasks: Inference and Learning

We consider six diffusion processes to compare the performance of CVI-DP and VDP. For both inference and learning, 5-fold cross-validation is performed. We discuss each diffusion process and the experimental setup in detail below.

Ornstein–Uhlenbeck We start with the (linear) Ornstein–Uhlenbeck diffusion process as a sanity check where the exact posterior and the exact log-likelihood are available in closed-form. It is defined as

$$\mathrm{d}x_t = -\theta x_t \, \mathrm{d}t + \mathrm{d}\beta_t, \tag{A119}$$

where β_t is standard Brownian motion ($Q_c = 1$). For the experiment, we set $\theta = 0.5$, $x_0 = 1$, $t_0 = 0$, $t_1 = 10$ and randomly observe 40 observations under the Gaussian likelihood with $\sigma^2 = 0.01$. To simulate the process, Euler–Maruyama is used with 0.01 step-size.

For inference, both the methods (CVI-DP and VDP) have an Ornstein–Uhlenbeck DP as prior with $\theta = 1.2$ and $Q_c = 1.0$. To get the (exact) posterior and the (exact) log-marginal likelihood, we use a Gaussian process regression (GPR) model with a matched Matérn-$1/2$ kernel. In the linear DP setup, CVI-DP does a single-step update with $\rho = 1$ for all values of the discretization grid ($\Delta t = \{0.01, 0.005, 0.001\}$) (theoretical reason of the single-step update is discussed in App. E.1). For VDP, we perform a grid-search over various learning-rate ω and use the best-performing one: 1.0 for $\Delta t = 0.001$, 0.5 for $\Delta t = 0.005$, and 0.1 for $\Delta t = 0.01$. As all the models converge to the same posterior, the posterior obtained by CVI-DP with $\Delta t = 0.01$ is only plotted in Fig. A8 along with the convergence plot for all the methods.

For learning, the same setup as in the inference experiment is used, but the parameter θ of the prior OU (lengthscale and variance of Matérn-$1/2$ kernel in GPR) is also optimized. For all the methods, θ is initialized
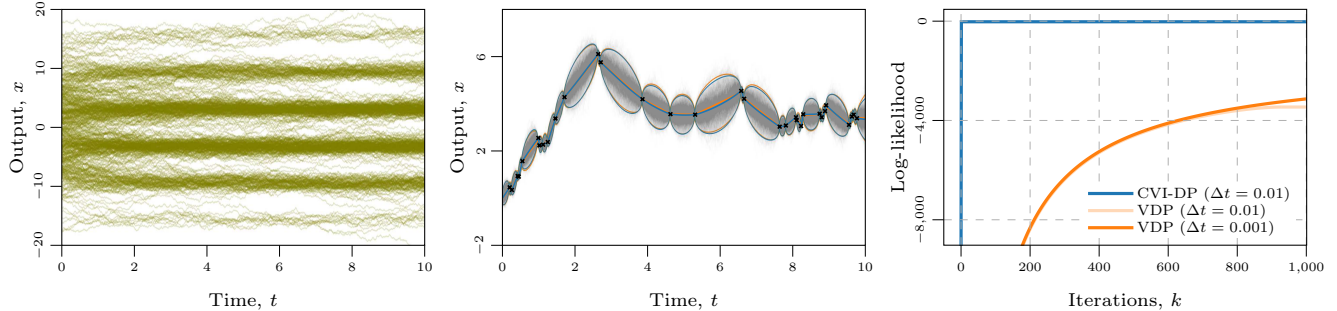
Figure A10: Approximate inference under a Sine prior (draws from prior on the left). Middle: Approximate posterior processes for CVI-DP and VDP overlaid on the SMC ground-truth samples. Right: CVI-DP converges quickly even with large discretization step when inferring the variational parameters, while VDP suffers from slow convergence even with a small discretization step.

to 2.5 (to be off from the expected optima) and the Adam optimizer is used. For the learning rate of Adam optimizer, we perform a grid search and use the best-performing one: 0.1 for CVI-DP and 0.01 for VDP.

Beneš We experiment with a commonly used non-linear diffusion process, the Beneš DP, whose marginal state distributions are bimodal and mode-switching in sample state trajectories becomes increasingly unlikely with time (Fig. A9). It is defined as:

$$\mathrm{d}x_t = \theta \tanh(x_t)\,\mathrm{d}t + \mathrm{d}\beta_t, \tag{A120}$$

where β_t is standard Brownian motion ($Q_\mathrm{c} = 1$). For the experiment, we set $\theta = 1.0$, $x_0 = 0$, $t_0 = 0$, $t_1 = 8$ and randomly observe 40 observations under the Gaussian likelihood with $\sigma^2 = 0.01$. To simulate the process, Euler–Maruyama is used with 0.01 step-size.

For inference, both the methods (CVI-DP and VDP) have a Beneš DP as prior with $\theta = 1.0$ and $Q_\mathrm{c} = 1.0$. After performing a grid search over the learning rate for both the methods, we use the best performing one: $\rho = 1$ in CVI-DP for all discretization grid ($\Delta t = \{0.01, 0.005, 0.001\}$), and $\omega = 0.1$ for $\Delta t = 0.001$, $\omega = 0.001$ for $\Delta t = \{0.005, 0.01\}$ for VDP. The posterior of both methods and the convergence plot are shown in Fig. A9. From the plot, it can be observed that the posterior obtained by both methods is identical. However, CVI-DP converges faster than VDP.

For learning, the same setup as in the inference experiment is used, and now the parameter θ of the prior Beneš DP is also optimized. For both methods, θ is initialized to 3 (to be off from the expected optima), and Adam optimizer is used. For the learning rate of Adam optimizer, we perform a grid search and use the best-performing one: 0.1 for CVI-DP and 0.01 for VDP.

Double-Well We experiment with the non-linear diffusion process, the Double-Well (DW) DP, whose marginal state distributions have two modes that sample state trajectories keep visiting through time (Fig. 3). It is defined as:

$$\mathrm{d}x_t = \theta_0\, x_t(\theta_1 - x_t^2)\,\mathrm{d}t + \mathrm{d}\beta_t, \tag{A121}$$

where β_t is standard Brownian motion ($Q_\mathrm{c} = 1$). For the experiment, we set $\theta_0 = 4.0$, $\theta_1 = 1.0$, $x_0 = 1.0$, $t_0 = 0$, $t_1 = 20$ and randomly observe 40 observations under the Gaussian likelihood with $\sigma^2 = 0.01$. To simulate the process, Euler–Maruyama is used with 0.01 step-size.

For inference, both the methods (CVI-DP and VDP) have a double-well DP as prior with $\theta_0 = 4.0$, $\theta_1 = 1.0$ and $Q_\mathrm{c} = 1.0$. After performing a grid search over the learning rate for both the methods, we use the best performing one: $\rho = 0.5$ for CVI-DP and $\omega = 0.001$ for VDP for all discretization grid ($\Delta t = \{0.01, 0.005, 0.001\}$). The posterior of both methods and the convergence plot are shown in Fig. 3. The plot shows that VDP struggles with convergence issues while CVI-DP converges faster and to a better ELBO value. With optimization tricks and more iterations, VDP is expected to reach the same posterior as CVI-DP.

For learning, the same setup as in the inference experiment is used, and now the parameter θ_1 of the prior DW DP is also optimized. For both the methods, θ_1 is initialized to 0.0 (to be off from the expected optima), and Adam optimizer is used. For the learning rate of Adam optimizer, we perform a grid search and use the best-performing
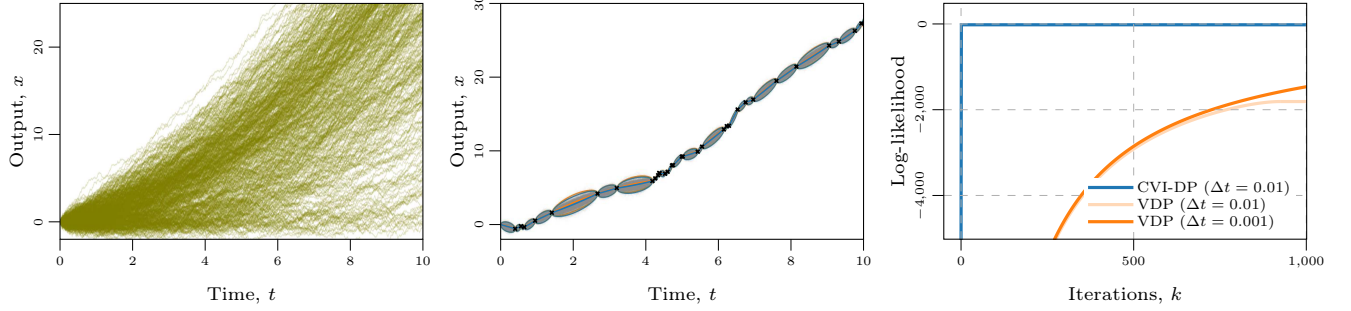
Figure A11: Approximate inference under a Sqrt prior (draws from prior on the left). Middle: Approximate posterior processes for CVI-DP and VDP overlaid on the SMC ground-truth samples. Right: CVI-DP converges quickly even with large discretization step when inferring the variational parameters, while VDP suffers from slow convergence even with a small discretization step.

one: 0.1 for CVI-DP and 0.01 for VDP. Fig. 5 showcases the fast learning of θ_1 in CVI-DP as compared to VDP.

Sine We experiment with the non-linear diffusion process, Sine DP, whose marginal state distributions have many modes (Fig. A10). It is defined as:

$$\mathrm{d}x_t = \theta_0 \sin(x_t - \theta_1)\,\mathrm{d}t + \mathrm{d}\beta_t, \tag{A122}$$

where β_t is standard Brownian motion ($Q_c = 1$). For the experiment, we set $\theta_0 = 1.0$, $\theta_1 = 0.0$, $x_0 = 0$, $t_0 = 0$, $t_1 = 10$ and randomly observe 40 observations under the Gaussian likelihood with $\sigma^2 = 0.01$. To simulate the process, Euler–Maruyama is used with 0.01 step-size.

For inference, both the models (CVI-DP and VDP) have a Sine DP as prior with $\theta_0 = 1.0$, $\theta_1 = 0.0$, and $Q_c = 1.0$. After performing a grid search over the learning rate for both the methods, we use the best performing one: $\rho = 1.0$ for CVI-DP and $\omega = 10^{-5}$ for VDP for all discretization grid ($\Delta t = \{0.01, 0.005, 0.001\}$). The posterior and convergence plot of both methods is shown in Fig. A10. From the plot, it can be observed that the posterior obtained by both methods is identical. However, CVI-DP converges faster than VDP. For VDP, in the experiment, we set the maximum iterations to 1000. However, with more iterations and optimization tricks, VDP is expected to reach the same ELBO value as CVI-DP. Empirically, we found that VDP suffers from slow convergence and takes ∼7000 iterations to lead to a value closer to CVI-DP.

For learning, the same setup as in the inference experiment is used, and now the parameter θ_1 of the prior Sine DP is also optimized. For both methods, θ_1 is initialized to 2.0 (to be off from the expected optima), and Adam optimizer is used. For the learning rate of Adam optimizer, we perform a grid search and use the best-performing one: 0.01 for both CVI-DP and VDP. While learning, after performing a grid search, the best-performing learning rate for CVI-DP was $\rho = 0.1$ while for VDP, it was the same as in inference, $\omega = 10^{-5}$.

Square-root We experiment with the non-linear diffusion process, Square-root DP, that has divergent fat-tailed behaviour (Fig. A11). It is defined as:

$$\mathrm{d}x_t = \sqrt{\theta|x_t|}\,\mathrm{d}t + \mathrm{d}\beta_t, \tag{A123}$$

where β_t is the standard Brownian motion ($Q_c = 1$). For the experiment, we set $\theta = 1.0$, $x_0 = 0.0$, $t_0 = 0$, $t_1 = 10$ and randomly observe 40 observations under the Gaussian likelihood with $\sigma^2 = 0.01$. To simulate the process, Euler–Maruyama is used with 0.01 step-size.

For inference, both the models (CVI-DP and VDP) have a Square-root DP as prior with $\theta = 1.0$ and $Q_c = 1.0$. After performing a grid search over the learning rate for both the methods, we use the best performing one: $\rho = 1.0$ for CVI-DP and $\omega = 10^{-5}$ for VDP for all discretization grid ($\Delta t = \{0.01, 0.005, 0.001\}$). The posterior and convergence plot of both methods is shown in Fig. A10. From the plot, it can be observed that the posterior obtained by both methods is identical. However, CVI-DP converges faster than VDP. For VDP, in the experiment, we set the maximum iterations to 1000. However, with more iterations and optimization tricks, VDP is expected to reach the same ELBO value as CVI-DP. Empirically, we found that VDP suffers from slow convergence and takes ∼7000 iterations to lead to a value closer to CVI-DP.

For learning, the same setup as in the inference experiment is used, and now the parameter θ of the prior Sine DP is also optimized. For both methods, θ is initialized to 5.0 (to be off from the expected optima), and Adam optimizer is used. For the learning rate of Adam optimizer, we perform a grid search and use the best-performing one: 0.01 for both CVI-DP and VDP. While learning, after performing a grid search, the best-performing learning rate for CVI-DP was $\rho = 0.5$ while for VDP, it was the same as in inference, $\omega = 10^{-5}$.

Stochastic van der Pol oscillator We experiment with the multi-dimensional non-linear diffusion process, stochastic van der Pol oscillator DP (Fig. 4). It is defined as:

$$\mathrm{d}\begin{bmatrix} x_t^{(1)} \\ x_t^{(2)} \end{bmatrix} = \theta_0 \begin{bmatrix} \theta_1\, x_t^{(1)} - (1/3)x_t^{(1)} - x_t^{(2)} \\ (1/\theta_1)x_t^{(1)} \end{bmatrix} \mathrm{d}t + \mathrm{d}\boldsymbol{\beta}_t, \tag{A124}$$

where β_t is the standard Brownian motion ($Q_c = \mathbf{I}$). For the experiment, we set $\theta_0 = 5.0$, $\theta_1 = 2.0$, $x_0 = 1.0$, $t_0 = 0$, $t_1 = 5$ and randomly observe 40 observations under the Gaussian likelihood with $\sigma^2 = 0.01\,\mathbf{I}$. To simulate the process, Euler–Maruyama is used with 0.01 step-size.

For inference, both the models (CVI-DP and VDP) have a stochastic van der Pol oscillator DP as prior with $\theta_0 = 5.0$, $\theta_1 = 2.0$ and $Q_c = \mathbf{I}$. After performing a grid search over the learning rate for both the methods, we use the best performing one: $\rho = 0.5$ for CVI-DP and $\omega = 10^{-3}$ for VDP for all discretization grid ($\Delta t = \{0.01, 0.005, 0.001\}$). The posterior and convergence plot of both methods is shown in Fig. 4. In the multi-dimensional setup, SMC struggles to converge and requires more iterations. Currently, for consistency, we use the same number of particles and smoothers for the SMC as other experiments. From the plot, it can be observed that CVI-DP converges faster than VDP. For VDP, in the experiment, we set the maximum iterations to 2000. However, with more iterations and optimization tricks, VDP is expected to reach the same ELBO value as CVI-DP.

For learning, the same setup as in the inference experiment is used, and now the parameters θ_0, θ_1 of the prior DP is also optimized. For both methods, θ_0, θ_1 are initialized to 2.0 (to be off from the expected optima), and Adam optimizer is used. For the learning rate of Adam optimizer, we perform a grid search and use the best-performing one: 0.01 for both CVI-DP and VDP. While learning, after performing a grid search, the best-performing learning rate for CVI-DP was $\rho = 0.5$ while for VDP, it was the same as in inference, $\omega = 10^{-3}$.

I.2 Finance Data

While the main focus is on providing a better method and algorithm for the particular case defined by VDP (and thus benchmark primarily against it), we also seek to showcase the practical applicability of our approach. To showcase the capability of proposed method CVI-DP in the real world, we experiment with a finance data set originally proposed for Student-t processes in Solin and Särkkä (2015). The data set considers the (log) stock price of Apple Inc. and consists of 8537 trading days (setup follows Solin and Särkkä, 2015). This experiment aims to learn the underlying process of the stock price, which is measured in terms of negative log predictive density (NLPD) on the hold-out test set. We bin the data into five bins, $\sim$1707 trading days in every bin and model the log stock price. We experiment with three models; each model has different prior information incorporated in it. Gaussian likelihood is used with $\sigma^2 = 0.25$ for all the models, which is not optimized.

Following Solin and Särkkä (2015), we consider a GP baseline: sparse variational Gaussian process (SVGP) with 500 inducing points (Titsias, 2009) in which prior information is incorporated in terms of the sum of kernels (Const.+Lin.+Matérn-$3/2$+Matérn-$1/2$). The prior is structured as in Solin and Särkkä (2015) to capture a linear trend, a slowly moving smooth trend component, and a faster more volatile component. The GP baseline gives NLPD 1.44±0.70 / RMSE 0.91±0.54. All the kernels are initialized with unit variance and lengthscale, and Adam optimizer is used with a learning rate of 0.1. The inducing variables are spread uniformly over the time grid and not optimized.

Next, we experiment with CVI-DP with a linear OU DP process in which prior is incorporated in terms of the OU process, which is initialized with $\theta=1.0, Q_c=0.1$. After evaluating the data, the prior on the initial state is set to $N(-1, 0.1)$. The learning rate ρ is set to 1.0 and the prior DP parameter is optimized using Adam optimizer with a 0.01 learning rate. The model gives NLPD 1.08±0.45 / RMSE 0.77±0.41.

Finally, to give more flexibility, we experiment with CVI-DP with a neural network drift (NN) DP. The drift of the prior DP f_p is initialized to be a NN with one hidden layer with three nodes followed by a ReLU activation function, and Q_c is set to 0.1. The parameters of the NN are initialized from a unit Gaussian, and after evaluating
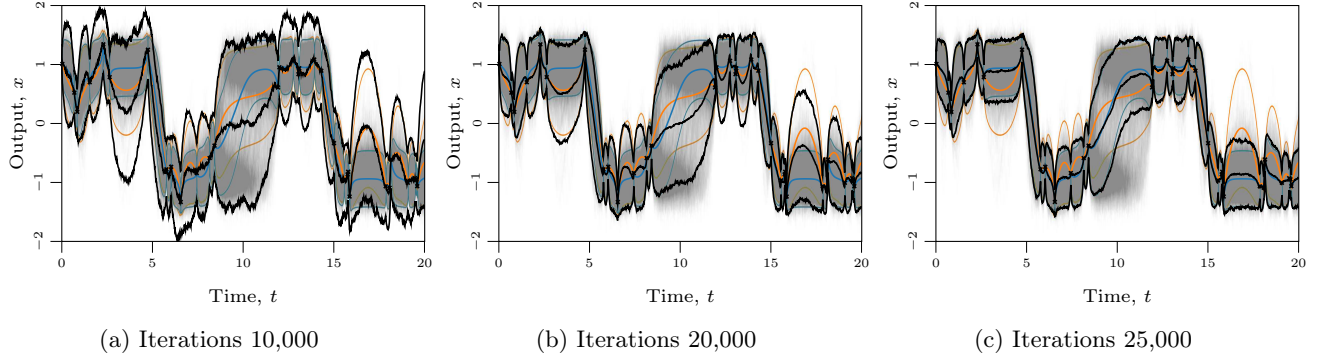
| (a) Iterations 10,000 | (b) Iterations 20,000 | (c) Iterations 25,000 |

Figure A12: Approximate posterior processes under a Double-Well prior (setup similar to Fig. 3) for ▬ CVI-DP, ▬ VDP, and ▬ NeuralSDE (Kidger et al., 2021) with different iterations overlaid on the SMC ground-truth samples. The NeuralSDE posterior gets closer to CVI-DP with more iterations.

the data, prior on the initial state is set to $N(-1, 0.1)$. The learning rate ρ is set to 0.5 and the prior DP parameter is optimized using Adam optimizer with 10^{-3}. The model gives NLPD 0.81±0.08 / RMSE 0.51±0.08.

We also experiment with VDP with a neural network drift (NN) DP. The drift of the prior DP f_p is initialized to be a NN with one hidden layer with three nodes followed by a ReLU activation function, and Q_c is set to 0.1. The parameters of the NN are initialized from a unit Gaussian, and after evaluating the data, prior on the initial state is set to $N(-1, 0.1)$. The learning rate ρ is set to 0.1 and the prior DP parameter is optimized using Adam optimizer with 10^{-3}. The model gives NLPD 0.92±0.22 / RMSE 0.54±0.17.

Of all the models, CVI-DP with a NN as drift results in the best NLPD / RMSE value as it is the most flexible and can adapt to the non-stationary behaviour of the data over the state range. To show the quality of learnt drift, we simulate and plot the predictions from the learnt prior DP for future years Fig. 6.

I.3 GPS Tracking Data

To further showcase the applicability of CVI-DP in real-world, we experiment with a GPS data set of a moving vehicle (data also from Solin and Särkkä (2015)). The aim is to model the 2D trajectory of the vehicle recorded from GPS coordinates over time. The data set is 106 minutes long and consists of 6373 observation points. We experiment with two models; both use two independent DPs learnt jointly (one for latitude and one for longitude directions). Similar to the setup in Solin and Särkkä (2015), we split the data in chunks of 30 s and perform 10-fold cross-validation. For all the models, Gaussian likelihood is used with $\sigma^2 = 0.01$, which is not optimized.

First, we experiment with CVI-DP with a linear OU DP process in which prior is incorporated in terms of the OU process, which is initialized with $\theta = 1.0, Q_c = 0.1$. After evaluating the data, prior on the initial state is set to $N(0, 0.1)$. For optimizing sites, the learning rate ρ is set to 0.5, and the prior DP parameter θ is optimized using Adam optimizer with a 0.01 learning rate. The model gives NLPD -0.67±0.19 / RMSE 0.13±0.04. Next, similar to the finance example (App. I.2), to give more flexibility, we experiment with CVI-DP with a neural network (NN) drift DP f_p. The drift of the DP is initialized to be a NN with one hidden layer with three nodes followed by a ReLU activation function, and Q_c is set to 0.1. The parameters of the NN are initialized from a unit Gaussian, and after evaluating the data, prior on the initial state is set to $N(0, 0.1)$. The learning rate ρ is set to 0.5 and the prior DP parameter is optimized using Adam optimizer with 10^{-2} learning rate. The model gives NLPD -0.82±0.43 / RMSE 0.06±0.03.

We also experiment with VDP with a neural network drift (NN) DP. The drift of the prior DP f_p is initialized to be a NN with one hidden layer with three nodes followed by a ReLU activation function, and Q_c is set to 0.1. The parameters of the NN are initialized from a unit Gaussian, and after evaluating the data, prior on the initial state is set to $N(0, 0.1)$. The learning rate ρ is set to 0.1 and the prior DP parameter is optimized using Adam optimizer with 10^{-3}. The model gives NLPD -0.55±0.24 / RMSE 0.09±0.06.

CVI-DP with a NN drift gives better NLPD and RMSE value primarily because of the flexibility of the DP to model the trajectory and possibility to adapt to the non-stationary behaviour in the state space. The posterior

for CVI-DP with NN drift is shown in Fig. 7. The behaviour of the vehicle is different at different points in the input space (perhaps due to faster driving on highways and slower on smaller streets), which the model is able to capture by learning the parameters.

I.4 Comparison with NeuralSDEs

We also compare against the recently popular class of NeuralSDE methods (Li et al., 2020a; Kidger et al., 2021). These methods are variational inference algorithms with a broader scope than CVI-DP: the posterior process q is not restricted to be a linear DP but is characterized by a neural network drift. However, they rely on sample-based simulation methods for estimation of the ELBO gradient and thus incur a large computational cost. Also, the convergence of optimization via stochastic gradient descent is often slow.

Empirically, we find that these methods need a delicate learning-rate scheduler to converge. In comparison, CVI-DP has a deterministic objective and intrinsically has an adaptive learning rate (setup similar to various natural gradient descent algorithms). Thus, CVI-DP is fast and does not need fine-tuning. Fig. A12 showcases the posterior of NeuralSDE along with CVI-DP and VDP (similar to Fig. 3). From the figure, it can be noted that in fewer iterations, the posterior is similar to VDP and with more iterations, the posterior gets closer to what CVI-DP gets. For the experiment, we use 1000 training samples for each iteration step and Adam optimizer with a 0.1 learning rate and an exponential learning-rate scheduler. The implementation is based on the code-base of Kidger et al. (2021).

Fig. A12 shows that while the NeuralSDE approach is general and eventually converges to the same optima as CVI-DP, in the particular case where it is possible to formalize the problem under the VDP/CVI-DP setting, using CVI-DP for inference and learning is orders of magnitude faster.

I.5 Implementation Detail

We implement VDP by solving the fixed point iterations using Euler's scheme. The experiments with CVI-DP are performed with the discrete version where Euler–Maruyama scheme is employed. The fixed point iterations of VDP can be solved via other adaptive ODE solvers. However, for a fair comparison, we would need to compare them with the continuous version of CVI-DP as discussed in App. G.

J Author Contributions

The initial idea and motivation of this work was conceived by VA in discussion with PV. PV had the main responsibility of implementing the method and conducting the experiments with help from AS and VA. The first draft was written by VA and PV. All authors contributed to finalizing the manuscript.