SMOOTHING NONLINEAR VARIATIONAL OBJECTIVES WITH SEQUENTIAL MONTE CARLO

Anonymous authors
Paper under double-blind review

ABSTRACT

The task of recovering nonlinear dynamics and latent structure from a population recording is a challenging problem in statistical neuroscience motivating the development of novel techniques in time series analysis. Recent work has focused on connections between Variational Inference and Sequential Monte Carlo for performing inference and parameter estimation on sequential data. Inspired by this work, we present a framework to develop Smoothed Variational Objectives (SVOs) that condition proposal distributions on the full time-ordered sequence of observations. SVO maintains both expressiveness and tractability by sharing parameters of the evolution term between the proposal and target. We apply the method to several dimensionality reduction/expansion tasks and examine the dynamics learned with a quantitative metric. SVO performs favorably against the state of the art.

1 INTRODUCTION

Conductance based models of excitable cells are widely used in neuroscience to describe the spiking activity of individual neurons. It is thought that neural computation is explained by dynamics and that these dynamics often exist in a lower or higher dimensionality than that of the recorded neural populations (Paninski & Cunningham, 2017). To extract the information encoded in firing activity, it is paramount to develop models that allow for tractable analysis of neural data. For example, experimentalists who have access to a single-dimensional observation such as a voltage recording are faced with the task of recovering the multidimensional nonlinear latent dynamics and trajectories through a higher dimensional space that describe the system of interest.

There is a large body of work for inferring latent trajectories for data governed by nonlinear dynamics or observations (Archer et al., 2015; Krishnan et al., 2015; Pandarinath et al., 2017; Diaz et al., 2019). Variational Inference (VI) and Markov Chain Monte Carlo (MCMC) are two popular approaches for performing inference in latent variable models. Recently connections have been established between both methods that allow for performing inference by defining a flexible variational family of filtered distributions using Sequential Monte Carlo (SMC) (Le et al., 2018; Maddison et al., 2017; Naesseth et al., 2018a). Unlike filtering, smoothing refers to the ability to condition states and parameters on the full time-ordered sequence of observations. In this framework, we sketch a method to construct Smoothed Variational Objectives (SVOs) using recursive backwards sampling algorithms (referred to as particle smoothers) in contrast to architectures that condition the encoding function on the full time ordered sequence of observations.

Recent literature argues that tighter variational bounds may hurt the proposal learning (Rainforth et al., 2018). More specifically, in the context of the Importance Weighted AutoEncoder (IWAE), the marginal likelihood estimate becomes exact as the number of particles $K \to \infty$. This can lead to a degenerated gradient estimate for the proposal with the signal-to-noise ratio shrinking at the rate $\mathcal{O}(\sqrt{1/K})$ (Rainforth et al., 2018). (Le et al., 2018) built on this idea to use different objectives for recognition and generative networks. Here we argue that the theoretical result in the IWAE setting cannot be directly extended to SVO due to the resampling process. We present empirical results to show that the degrading signal-to-noise ratio is addressed by the choice of biased gradient estimators.

The contributions of this paper are highlighted below. First, we describe a factorized proposal distribution that (i) disentangles the transition terms from the terms that represent the encoding of the data and (ii) shares variational parameters of the transition function with the target distribution. We present results illustrating the positive effect of the number of particles on the signal-to-noise ratio for the transition, encoder and decoder parameters within our factorization. An important way to ascertain the quality of the learned dynamics is to use the transition terms in the generative model to propagate the system forwards without input data and then to make observation predictions. We show that SVO obtains an improved estimate of the latent states as measured by its ability to more accurately predict observations using the dynamics learned.

The remainder of this paper is organized as follows. Section 2 summarizes related work on VI and SMC. Section 3 gives preliminaries and defines the proposal introducing three types of gradient estimators. We then present results on both filtered and smoothed objectives with two benchmark datasets and single cell electrophysiology data from the Allen Institute in Section 4. Section 5 concludes.

2 Related Work

There is a large body of work on fitting State Space Models (SSMs) of increasing complexity using VI. These methods can be separated into two classes. Pure VI methods optimize a lower bound to the log likelihood parameterized by defining a variational family. Hybrid methods use SMC to construct an estimate for the log marginal likelihood and optimize a surrogate lower bound. We review the details of both techniques in the preliminaries section and outline the contributions of related methods below.

Gaussian Process Factor Analysis (Yu et al., 2009) (GPFA) is a statistical model that assumes linear time invariant transition and emission functions. GfLDS (Gao et al., 2016; Archer et al., 2015) is a VI scheme that extends the above by positing a linear dynamical system with nonlinear emission densities. VIND (Diaz et al., 2019) is a VI scheme that permits both transition and emission densities to be represented as arbitrary nonlinearities. It is worth noting that nonlinear evolution in the latent space introduces an additional inference challenge. Marginalizing latent variables at time t produces an intractable latent factor for the previous time point. VIND maintains tractability of the variational approximation by forming a Laplace approximation to an intractable non Gaussian parent approximate distribution. The mean and the covariance of the Laplace approximation are obtained via the Fixed Point Iteration (FPI) and the intractable terms are reused within the approximation. GPFA, GfLDS and VIND are all smoothers in that they use the full sequence of observations to infer the current latent state.

Deep Kalman Filters (DeKF) (Krishnan et al., 2015) are an inference method for nonlinear-state space models. Linear functions of the Kalman filter are replaced with deep networks. The variational posterior is designed to exploit the fact that latent variables in the true posterior depend only on future observations. This idea is developed in (Krishnan et al., 2016) by proposing an approximation that shares conditional structure with the generative model.

Latent Factor Analysis via Dynamical Systems (LFADS) (Pandarinath et al., 2017; Sussillo et al., 2016) is a bidirectional Deep-Learning based RNN architecture designed for neural data anaysis. The design choice in LFADS incorporates information from past and future observations via the network architecture as opposed to performing message passing. Unlike VI methods that use a Kalman filter styled covariance matrix (Krishnan et al., 2015; 2016; Archer et al., 2015; Gao et al., 2016), the transition functions in LFADS are deterministic.

An approach to increase model capacity is the development of tighter bounds than the ELBO. Filtering Variational Objectives (Maddison et al., 2017) (FIVO) and Variational Sequential Monte Carlo (Naesseth et al., 2018a) are related methods that use SMC to construct a filtered objective function for VI. Auto-Encoding Sequential Monte Carlo (Le et al., 2018) (AESMC) is a similar method to learn both proposal distributions for SMC that act as variational approximations and generative models using deep neural networks. Unlike GfLDS and VIND, these methods do not require inverting a block-tridiagonal matrix which mixes components of state space through the inverse covariance. Information from the complete time ordered sequence $1 : T$ is not applied directly to the transition function to the future of the current time point $t < T$ to infer the latent state.

3 METHODS

3.1 PRELIMINARIES

3.1.1 VI AND IMPORTANCE WEIGHTED AUTO-ENCODERS

Inference in latent variable models requires marginalizing a generative distribution with respect to hidden variables $\mathbf{Z}$.

$$\log p_\theta(\mathbf{X}) = \int \log p_\theta(\mathbf{X}, \mathbf{Z}) d\mathbf{Z} \tag{1}$$

VI describes family of techniques for approximating a solution to the above. The idea is to define a tractable distribution $q_\phi(\mathbf{Z}|\mathbf{X})$ and to optimize a lower bound to the log likelihood:

$$\log p_\theta(\mathbf{X}) \geq \mathcal{L}_{\text{ELBO}}(\theta, \phi, \mathbf{X}, \mathbf{Z}) = \mathop{\mathbb{E}}_{q}\left[\log \frac{p_\theta(\mathbf{X}, \mathbf{Z})}{q_\phi(\mathbf{Z}|\mathbf{X})}\right] \tag{2}$$

Tractability and expressiveness of the variational approximation $q_\phi(\mathbf{Z}|\mathbf{X})$ are contrasting goals. The distribution $q_\phi(\mathbf{Z}|\mathbf{X})$ must be chosen carefully; a poor choice of the variational family defining $q_\phi(\mathbf{Z}|\mathbf{X})$ will affect the generative model by pulling the parameters θ towards ϕ.

$$\theta^*, \phi^* = \mathop{\arg\min}_{\theta\in\Theta, \phi\in\Phi} D_{\text{KL}}(q_\phi || p_\theta) = \mathop{\arg\max}_{\theta\in\Theta, \phi\in\Phi} \mathcal{L}_{\text{ELBO}} \tag{3}$$

Auto Encoding Variational Bayes (Kingma & Welling, 2013) (AEVB) is method to simultaneously encode and decode the parameters $\{\phi, \theta\}$ for both $q_\phi(\mathbf{Z}|\mathbf{X})$ and $p_\theta(\mathbf{X}, \mathbf{Z})$. When the densities are not in the natural exponential family, it is possible to draw samples from $q_\phi(\mathbf{Z}|\mathbf{X})$ to evaluate $\mathbb{E}_{q_\phi}[\log p_\theta(\mathbf{X}, \mathbf{Z}) - \log q_\phi(\mathbf{Z}|\mathbf{X})]$ numerically. We can approximate both $\mathcal{L}_{\text{ELBO}}$ terms reparameterizing in order to compute the gradients (Kingma & Welling, 2013; Rezende et al., 2014) as follows:

$$\mathcal{L}_{\text{AEVB}}(\theta, \phi, \mathbf{X}^{(i)}) \approx \frac{1}{L}\sum_{l=1}^{L} \log \frac{p_\theta(\mathbf{X}^{(i)}, \mathbf{Z}^{(i,l)})}{q_\phi(\mathbf{Z}^{(i,l)}|\mathbf{X}^{(i)})} \tag{4}$$

$$\text{where} \quad \mathbf{Z}^{(i,l)} = \mu_\phi + \Sigma_\phi^{1/2}\epsilon^{(i,l)} \quad \text{and} \quad \epsilon^{(i,l)} \sim \mathcal{N}(0, 1) \tag{5}$$

Building upon this, the Importance Weighted Auto Encoder (Burda et al., 2015) (IWAE) constructs tighter bounds than the $\mathcal{L}_{\text{AEVB}}$ through mode averaging as opposed to mode matching. The idea is to take an importance sampling estimate of $\mathcal{L}_{\text{AEVB}}$ by drawing K samples from a proposal distribution and to average probability ratios:

$$\mathcal{L}^K_{\text{IWAE}}(\theta, \phi, \mathbf{X}, \mathbf{Z}) \approx \log \frac{1}{K}\sum_{k=1}^{K} \frac{p_\theta(\mathbf{X}, \mathbf{Z}^{(k)})}{q_\phi(\mathbf{Z}^{(k)}|\mathbf{X})} \tag{6}$$

$$\text{where} \quad \mathbf{Z}^{(k)} \sim q_\phi(\mathbf{Z}|\mathbf{X}) \tag{7}$$

It can be shown that $\log p_\theta(\mathbf{X}) \geq \mathcal{L}^{k+1} \geq \mathcal{L}^k \geq \mathcal{L}^1$. The $\mathcal{L}_{\text{IWAE}}$ reduces to the $\mathcal{L}_{\text{AEVB}}$ when $K = 1$ and approaches the true log probability of the data as $K \to \infty$.

3.1.2 STATE SPACE MODELS

State space models (SSMs) describe a time series of observations $\mathbf{X} \equiv \{\mathbf{x}_1, \dots \mathbf{x}_T\}$, $\mathbf{x}_t \in \mathbb{R}^{d_X}$ dependent upon a time series of latent variables $\mathbf{Z} \equiv \{\mathbf{z}_1, \dots \mathbf{z}_T\}$, $\mathbf{z}_T \in \mathbb{R}^{d_Z}$ that evolve according to stochastic dynamics. We are interested models with the following dependency:

$$p_\theta(\mathbf{X}, \mathbf{Z}) = c_\theta \cdot F_\theta(\mathbf{Z}) \prod_{t=0}^{T} g_\theta(\mathbf{x}_t|\mathbf{z}_t)\,, \tag{8}$$

where g_θ is an observation model whose parameters are functions of the latent state $\mathbf{z}_t$. The normalization constant is denoted $c_{\phi,\theta}$ and F_ϕ denotes the latent evolution described with Markov Chain:

$$F_\theta(\mathbf{Z}) = f_0(\mathbf{z}_0) \prod_{t=1}^{T} f_\theta(\mathbf{z}_t|\mathbf{z}_{t-1})\,, \tag{9}$$

$$f_0 = \mathcal{N}\big(\psi_0, \mathbf{Q}_0\big)\,, \tag{10}$$

$$\mathbf{z}_t|\mathbf{z}_{t-1} \sim \mathcal{N}\big(\psi_\theta(\mathbf{z}_{t-1}),\, \mathbf{Q}\big)\,, \tag{11}$$

Marginalization with respect to $\mathbf{z}$ becomes intractable when $F_\theta(\mathbf{Z})$ is defined by a nonlinear function or $G(\mathbf{X}|\mathbf{Z})$ is non-Gaussian.

3.1.3 SEQUENTIAL MONTE CARLO

Sequential Monte Carlo (SMC) methods factorize an intractable distribution $p_\theta(\mathbf{X}, \mathbf{Z})$ (referred to as the target distribution) into a distribution of increasing probability spaces. Sequential Importance Sampling (SIS) methods perform importance sampling sequentially by defining importance weights $w_t^{(k)}$ at time t for sample k as follows

$$w_t^{(k)} \equiv \frac{f_\theta(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{(k)}) g_\theta(\mathbf{x}_t|\mathbf{z}_t^{(k)})}{q_\phi(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{(k)}, \mathbf{x}_t)} \tag{12}$$

where K samples are drawn from the proposal distribution q:

$$\mathbf{z}_t^{(k)} \sim q_\phi(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{(k)}, \mathbf{x}_t) \tag{13}$$

Various resampling schemes exist so that the samples which are referred to as particles are focused on promising regions of state space. SMC achieves this goal by resampling particles according to their importance sampling weights:

$$\mathbf{a}_{t-1}^{(k)} \sim \text{Discrete}(\cdot|w_{t-1}^{(1)}, \cdots, w_{t-1}^{(K)}) \tag{14}$$

$$w_t^{(k)} \equiv \frac{f_\theta(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{\mathbf{a}_{t-1}^{(k)}}) g_\theta(\mathbf{x}_t|\mathbf{z}_t^{\mathbf{a}_{t-1}^{(k)}})}{q_\phi(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{\mathbf{a}_{t-1}^{(k)}}, \mathbf{x}_t)} \tag{15}$$

At the last time step, the posterior can be evaluated by averaging over sample trajectories to approximate the functional integral.

$$\sum_{k=1}^{K} \bar{w}_T^{(k)} \delta_{\mathbf{z}_{1:T}^{(k)}}(\mathbf{z}_{1:T}) \tag{16}$$

where $\bar{w}_T^{(k)} = w_T^{(k)} / \sum_{j=1}^{K} w_T^{(j)}$. The weights can now be used to construct an unbiased estimate for the marginal likelihood:

$$\hat{\mathcal{Z}}_{\text{SMC}} \equiv \prod_{t=1}^{T} \Big[\frac{1}{K} \sum_{k=1}^{K} w_t^{(k)} \Big] \tag{17}$$

3.1.4 AUTO ENCODING SEQUENTIAL MONTE CARLO

An important insight of (Maddison et al., 2017; Le et al., 2018) is that the SMC algorithm is deterministic conditioning on $(\mathbf{Z}_{1:T}^{(1:K)}, \mathbf{A}_{1:T-1}^{(1:K)})$. As a result, the proposal can be reparameterized to act as a variational distribution that can be encoded:

$$Q_{\text{SMC}}(\mathbf{Z}_{1:T}^{1:K}, \mathbf{A}_{1:T-1}^{1:K}) \equiv \left(\prod_{k=1}^{K} q_{1,\phi}(\mathbf{z}_1^{(k)}) \right) \left(\prod_{t=2}^{T} \prod_{k=1}^{K} q_{t,\phi}(\mathbf{z}_t^{(k)}|\mathbf{z}_{1:t-1}^{\mathbf{a}_{t-1}^{(k)}}) \cdot \text{Discrete}(\mathbf{a}_{t-1}^{(k)}|\mathbf{w}_{t-1}^{1:K}) \right) \tag{18}$$

This gives a way of constructing a filtered cost function for simultaneous model inference and learning. The cost is constructed by running SMC to obtain an estimate of the marginal log likelihood. As with the IWAE, increasing K yields a tighter bound $\mathcal{L}_{\text{SMC}}$ defined below.

$$\mathcal{L}_{\text{SMC}}(\theta, \phi, \mathbf{Z}_{1:T}) \equiv \int d\mathbf{Z}_{1:T}^{1:N} d\mathbf{A}_{1:T}^{1:K} Q_{\text{SMC}}(\mathbf{Z}_{1:T}^{1:K}, \mathbf{A}_{1:T-1}^{1:K}) \log \hat{\mathcal{Z}}_{\text{SMC}}(\mathbf{Z}_{1:T}^{1:K}, \mathbf{A}_{1:T-1}^{1:K}) \tag{19}$$

3.2 SMOOTHED VARIATIONAL OBJECTIVES

We build upon AESMC and FIVO to design an ELBO based on the marginal likelihood constructed using SMC. With access to the entire sequence of observations $\mathbf{x}_{1:T}$, a Smoothed Variational Objective (SVO) is less vulnerable to variance in the observations $\mathbf{x}_t$ and is able to obtain a more accurate estimate of $\mathbf{z}_t$ at each time step.

3.2.1 PARAMETERIZING THE PROPOSAL DISTRIBUTION

We begin by considering a proposal distribution of the form:

$$q_{\phi,\varphi}(\mathbf{z}_{1:T}^{(k)}|\mathbf{x}_{1:T}) \propto \underbrace{q_{\phi,\varphi}(\mathbf{z}_1^{(k)})}_{initial\ state} \prod_{t=1}^{T} \underbrace{q_{\phi}(\mathbf{z}_t^{(k)}|\mathbf{x}_t)}_{encoding} \prod_{t=2}^{T} \underbrace{\text{Categorical}(\mathbf{a}_{t-1}^{(k)}|\mathbf{w}_{t-1}^{1:K})}_{resampling} \underbrace{q_{\phi,\varphi}(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{a_{t-1}^{(k)}})}_{evolution} \tag{20}$$

where the proposal density factorizes into separate functions for evolution of the latent dynamics and encodings of the data.

$$q_{\phi,\varphi}(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\psi(\mathbf{z}_{t-1}), \Sigma), \tag{21}$$

$$q_{\phi}(\mathbf{z}_t|\mathbf{x}_t) = \mathcal{N}(\gamma(\mathbf{x}_t), \Lambda). \tag{22}$$

We take $\psi : \mathbb{R}^{d_z} \to \mathbb{R}^{d_z}$ and $\gamma : \mathbb{R}^{d_x} \to \mathbb{R}^{d_z}$ as nonlinear time invariant functions represented with deep neural networks. The covariances Σ and Λ can be invariant trainable parameters or non-linear functions of the latent space. The respective means of Gaussian functions can be convolved retaining Gaussian functional form. Unlike Q_{SMC} in equation 18, SVO posits that the dynamics exist independent of time as a function of latent space. This proposal choice is also advantageous because the transition term of the recognition model $q_{\varphi}(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{(k)})$ can be chosen to share the parameters $\varphi = \{\psi, \Sigma\}$ with the target transition term $f_{\theta,\varphi}(\mathbf{z}_t|\mathbf{z}_{t-1})$ of the generative model.

$$p_{\theta,\varphi}(\mathbf{z}_{1:T}, \mathbf{x}_{1:T}) \propto \underbrace{f_{\theta,\varphi}(\mathbf{z}_1)}_{initial\ state} \prod_{t=1}^{T} \underbrace{g_{\theta}(\mathbf{x}_t|\mathbf{z}_t)}_{decoding} \prod_{t=2}^{T} \underbrace{f_{\theta,\varphi}(\mathbf{z}_t|\mathbf{z}_{t-1})}_{evolution} \tag{23}$$

This is analogous to the bootstrap filter, however the addition of a new term permits disentangling the transition function from an encoding of the data. As a result the evolution term of the variational posterior is exact retaining both tractability and expressiveness. The generative evolution law is thus specified as follows:

$$f_{\varphi}(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\psi(\mathbf{z}_{t-1}), \Sigma), \tag{24}$$

The decoding term is defined using a deterministic nonlinear rate function $\upsilon : \mathbb{R}^{d_z} \to \mathbb{R}^{d_x}$ represented with a deep network and a noise model that need not be conjugate. Without loss of generality consider a Gaussian emission density:

$$g_{\theta}(\mathbf{z}_t|\mathbf{x}_t) = \mathcal{N}(\upsilon(\mathbf{x}_t), \Gamma), \tag{25}$$

3.2.2 FORWARD FILTERING BACKWARD SMOOTHING

It is important to use the full time ordered sequence of observations $\mathbf{x}_{1:T}$ to obtain the best estimate of the initial state $\mathbf{z}_1$. We consider two solutions, one is a bidirectional encoding architecture for smoothing and another is the Forward Filtering Backwards Smoothing (FFBS) formula (Kitagawa, 1996). To smooth the latent states and ensure that $q(\mathbf{z}_t)$ depends on both $\mathbf{z}_{t-1}$ and $\mathbf{z}_{t+1}$, we consider the following recursive formula:

$$p(\mathbf{z}_t|\mathbf{x}_{1:T}) = \underbrace{p(\mathbf{z}_t|\mathbf{x}_{1:t})}_{\text{filtered}} \int \underbrace{\frac{\overbrace{p(\mathbf{z}_{t+1}|\mathbf{x}_{1:T})}^{\text{smoothed}} \overbrace{p(\mathbf{z}_{t+1}|\mathbf{z}_t)}^{\text{transition}}}{\int p(\mathbf{z}_{t+1}|\mathbf{z}_t) p(\mathbf{z}_t|\mathbf{x}_{1:t}) d\mathbf{z}_t}}_{\text{state prediction}} d\mathbf{z}_{t+1} \tag{26}$$

FFBS first iterates forward to compute the filtered distribution at each time step. Then it reweights particles with the following backward recursion:

$$w_{t|T}^{(k)} = w_t^{(k)} \left[\sum_{i=1}^{K} w_{t+1|T}^{(i)} \frac{f_{\theta}(\mathbf{z}_{t+1}^{(i)}|\mathbf{z}_t^{(k)})}{\sum_{j=1}^{K} w_t^{(j)} f_{\theta,\varphi}(\mathbf{z}_{t+1}^{(i)}|\mathbf{z}_t^{(j)})} \right] \tag{27}$$

with $w_{T|T}^{(k)} = w_T^{(k)}$, to obtain an approximated smoothed marginal likelihood,

$$\hat{\mathcal{Z}}_{\text{FFBS}} \equiv \prod_{t=1}^{T} \Big[\frac{1}{K} \sum_{k=1}^{K} w_{t|T}^{(k)} \Big]. \tag{28}$$

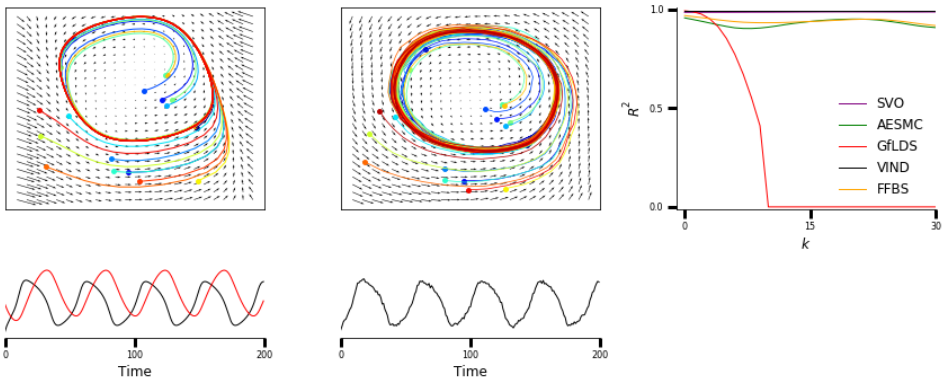

Figure 1: Summary of the Fitzhugh-Nagumo results: (top left) ground truth dynamics and trajectories for the original system; (bottom left) latent variables $\mathbf{z}_t = (V_t, W_t)$ from a single trial in the plane; (top center) latent dynamics and trajectories inferred by SVO; (bottom center) noisy 1D observation $\mathbf{x}_t = \mathcal{N}(V_t, \sigma^2)$ from a single trial; (right) R_k^2 for various models on the dimensionality expansion task.

We can now design an ELBO similar to equation 18 by replacing $\hat{\mathcal{Z}}_{\text{SMC}}$ with $\hat{\mathcal{Z}}_{\text{FFBS}}$.

When applying FFBS, we find two drawbacks. In contrast to the $\mathcal{O}(TK)$ complexity of particle filter, the complexity of FFBS is $\mathcal{O}(TK^2)$, scaling quadratically with the number of particles through the through the (i, k) pairwise interactions. Hence, it is expensive to optimize. In our experiments we found that when θ, ϕ are far from optimized values, the backward smoothing magnifies the error and harms the model learning. As a solution, we replace FFBS with the following smoothing method.

3.2.3 Encoding Architectures for Smoothed Variational Objectives

A natural approach to constructing a particle smoother is to condition the proposal distribution on the full time ordered sequence of observations. Redefining the proposal in equation 20:

$$q_{\phi,\varphi}(\mathbf{z}_{1:T}^{(k)}|\mathbf{x}_{1:T}) \propto q_{\phi,\varphi}(\mathbf{z}_1^{(k)}|\mathbf{x}_{1:T}) \prod_{t=1}^{T} q_\phi(\mathbf{z}_t^{(k)}|\mathbf{x}_{1:T}) \prod_{t=2}^{T} \text{Categorical}(\mathbf{a}_{t-1}^{(k)}|\mathbf{w}_{t-1}^{1:K}) q_{\phi,\varphi}(\mathbf{z}_t^{(k)}|\mathbf{z}_{t-1}^{a_{t-1}^{(k)}}) \tag{29}$$

For the case where $q_{t,\phi}(\mathbf{z}_1|\mathbf{x}_1)$ and $q_{t,\phi}(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \mathbf{x}_{1:t})$ are factorized Gaussians with conditionals parameterized by time varying networks, simply expanding the network inputs from $\mathbf{x}_t$ to $\mathbf{x}_{1:T}$ would be expensive. As a result, we experiment using architectures to extract features from the full time series $\mathbf{x}_{1:T}$ and redefine the input to our encoder functions. To extract features an RNN is run both forward and backward,

$$e_{f,t} = \text{RNN}_f(e_{f,t-1}, \mathbf{x}_t) \tag{30}$$

$$e_{b,t} = \text{RNN}_b(e_{b,t+1}, \mathbf{x}_t) \tag{31}$$

where $e_{f,t}$ and $e_{b,t}$ are the bidirectional RNN's states containing information of $\mathbf{x}_{1:t}$ and $\mathbf{x}_{t:T}$ respectively. Then the concatenation of $[e_{f,T}, e_{b,1}]$ is used to parameterize the initial state distribution $q_\phi(\mathbf{z}_1|\mathbf{x}_{1:T})$, and $[e_{f,t}, e_{b,t}]$ is fed into the term of the proposal defining the encoder $q_\phi(\mathbf{z}_t|\mathbf{x}_{1:T})$. The nonlinear time invariant evolution function is then applied to transition between states.

$$\mathbf{z}_1 = \gamma([e_{f,T}, e_{b,1}]) \tag{32}$$

$$\mathbf{z}_t = \psi(\mathbf{z}_{t-1}) \tag{33}$$

The cost of the feature extraction with the bidirectional RNN is $\mathcal{O}(T)$, independent of the number of particles unlike the FFBS.

3.2.4 Nonlinear Evolution of the Covariance Matrix.

For many nonlinear dynamical systems, the evolution of the latent state may depend on a noise term which itself may evolve stochastically. In order to model a covariance matrix that expresses

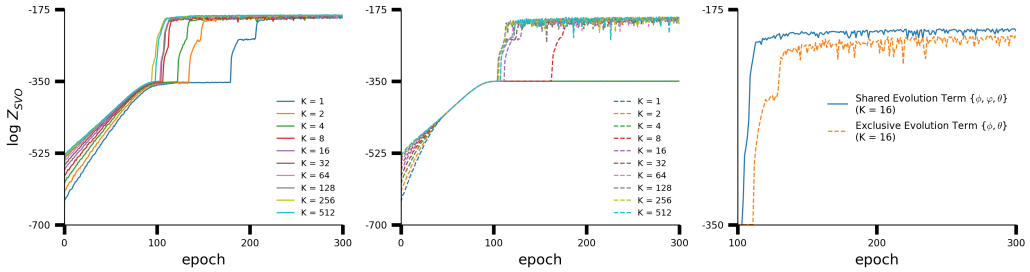

Figure 2: ELBO convergence across epochs for SVO using exclusive parameters θ, ϕ and shared parameters θ, φ, ϕ; (left) log $\mathcal{Z}_{SVO}$ across epochs as K increases using shared evolution network; (center) log $\mathcal{Z}_{SVO}$ across epochs as K increases using independent evolution networks; (right) log $\mathcal{Z}_{SVO}$ convergence for shared vs independent evolution networks with $K = 16$ highlighting faster convergence to a higher ELBO.

nonlinear $\mathbf{z}$-dependence on the latent space, we face a challenge. It is important to ensure that the latent paths are smooth. This is equivalent to stating that the difference between the covariance matrix and a constant matrix $\mathbb{C}$ is small.

$$\max |\mathbf{Q}(\mathbf{z}_t) - \mathbb{C}| \lesssim 0.1 \tag{34}$$

An expressive model should permit the covariance to undergo nonlinear evolution. One solution is to parameterize the covariance with a constant plus a scalar times a symmetric matrix whose components are a nonlinear function of the latent state.

$$\mathbf{Q}(\mathbf{z}_t) = \mathbb{C} + \alpha \cdot \mathbf{\Sigma}(\mathbf{z}_t) \tag{35}$$

In the experiments we take $\alpha = 1e-1$, $\mathbb{C} = \mathbb{I} \cdot \sigma^2$ where σ^2 is a trainable variable and the components of $\mathbf{\Sigma}(\mathbf{z}_t)$ as the output of a deep network. This permits dynamic proposal distributions whose entropy evolves as the Markov chain transitions between states. At different positions in latent space, the system can suppress or enhance its sensitivity to noise. We refer to this as a Locally Linear Covariance Matrix (LLCM).

3.3 Gradient Estimators

Assuming all $\mathbf{z}_t^{(k)}$ are reparameterized, we derive the gradient expression (see (Maddison et al., 2017; Le et al., 2018)):

$$\nabla \text{ELBO}_{\text{SVO}} = \mathbb{E}\left[\nabla \log \prod_{t=1}^{T-1} \prod_{k=1}^{K} \textsc{Categorical}(a_t^{(k)} | \mathbf{w}_t^{1:K}) \cdot \mathcal{L}_{\text{SVO}}^K + \nabla \mathcal{L}_{\text{SVO}}^K\right] \tag{36}$$

We consider the following three stochastic gradient estimators:

1. The Categorical resampled estimator,
   $\nabla \mathcal{L}_K$ + Categorical = $\nabla \log \prod_{t=1}^{T-1} \prod_{k=1}^{K} \textsc{Categorical}(a_t^{(k)} | \mathbf{w}_t^{1:K}) \cdot \mathcal{L}_{\text{SVO}}^K + \nabla \mathcal{L}_{\text{SVO}}^K$
2. The biased gradient estimator without resampling,
   $\nabla \mathcal{L}_{\text{SVO}}^K$
3. The relaxed gradient estimator,
   $\nabla \mathcal{L}_K$ + Concrete = $\nabla \log \prod_{t=1}^{T-1} \prod_{k=1}^{K} \textsc{Concrete}(a_t^{(k)} | \mathbf{w}_t^{1:K}, \lambda) \cdot \mathcal{L}_{\text{SVO}}^K + \nabla \mathcal{L}_{\text{SVO}}^K$,
   where $\textsc{Concrete}(\cdot | \mathbf{w}_t^{1:K}, \lambda) = (K-1)! \lambda^{K-1} \prod_{k=1}^{K} \left(w_t^{(k)} (a_t^{(k)})^{-\lambda-1} / \sum_{i=1}^{K} w_t^{(i)} (a_t^{(i)})^{-\lambda} \right)$
   is a relaxation of Categorical (Jang et al., 2016), (Maddison et al., 2016).

Note that as $\lambda \to 0$, the Concrete distribution approaches the Categorical. Clearly, the first gradient estimator is unbiased while the rest two are biased. We find that the gradient term accounting for the resampling step contributes to the majority of variance in the Categorical resampling gradient estimator, often increasing by the order of $\mathcal{O}(K)$. We observe a similar effect with the relaxed gradient estimator if we choose $\lambda \leq (K-1)^{-1}$. By contrast, the gradient estimator without resampling and the relaxed gradient estimator with moderate temperature are much more stable

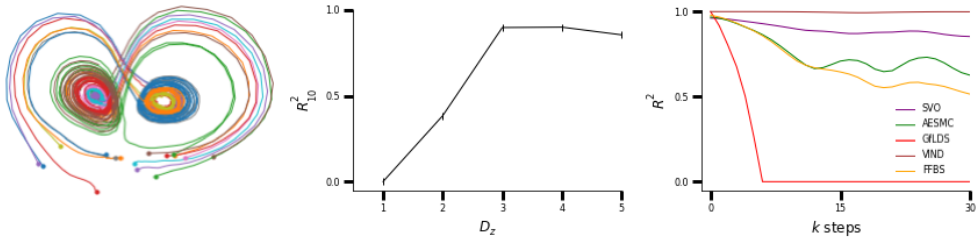

Figure 3: Summary of the Lorenz results: (left) inferred latent paths from noisy 10D observations; (center) the $R^2_{k=10}$ plot for ten step forward interpolation as latent dimension D_z increases; (right) the R^2_k for $D_z = 3$ across various models on the dimensionality reduction task.

despite the sacrifice of unbiasedeness; Furthermore, these two choices appear to give a promising learning performance for both proposal and generative networks with larger numbers of particles. We will formalize this idea as the signal-to-noise ratio and add empirical evidence in Section 4. For the rest of experiments, we use $\nabla \mathcal{L}^K_{\text{SVO}}$ to estimate the gradient.

4 Experimental Results

It is important to quantify the performance of the hidden evolution in order to understand the quality of the dynamics learned. The k-step MSE and its normalized version, the R^2_k are computed by applying the transition function to the system over a rolling window k steps into the future without any input data. The emission function is then used to form a predicted reconstruction $\hat{\mathbf{x}}_{t+k}$ and an error term using the corresponding observation window $\mathbf{x}_{t+k}$ (below $\bar{\mathbf{x}}_k$ is the average of $\mathbf{x}_{k:T}$). This ensures that the evaluation criteria is dependent upon the both evolution term as well as the emission term.

$$\text{MSE}_k = \sum_{t=0}^{T-k} \left(\mathbf{x}_{t+k} - \hat{\mathbf{x}}_{t+k}\right)^2 \qquad R^2_k = 1 - \frac{\text{MSE}_k}{\sum_{t=0}^{T-k} \left(\mathbf{x}_{t+k} - \bar{\mathbf{x}}_k\right)^2} \tag{37}$$

4.1 Fitzhugh-Nagumo

The Fitzhugh-Nagumo system is a two dimensional reduction of the Hodgkin-Huxley model. It is described by two independent variables with cubic and linear functions.

$$\begin{aligned} \dot{V} &= V - V^3/3 - W + I_{ext}\,, \\ \dot{W} &= a(bV - cW) \end{aligned} \tag{38}$$

The system was integrated over 200 time points with $I_{ext} = 1$ held constant and $a = 0.7, b = 0.8, c = 0.08$. The initial state was sampled uniformly over $[-3, 3]^2$ and 100 trials were generated with 66 for training, 17 for validation and 17 for testing. One dimensional Gaussian observations were produced with a linear observation model acting on a single variable. Dimensionality expansion is intrinsically harder than dimensionality reduction due to the loss of information. For this reason we choose $\mathbf{z}_t = (V_t, W_t)$ and $\mathbf{x}_t = \mathcal{N}(V_t, \sigma^2)$ where $\sigma^2 = 0.01$.

Fig. 1 shows the results of the Fitzhugh-Nagumo experiment. The top left panel displays the phase space and trajectories of the original system. The bottom left panel displays latent variables from a single trial in the plane. The top center panel displays the inferred two-dimensional dynamics and trajectories. We find that the limit cycle is recovered correctly and the topology of the space is preserved. Initial points located both inside and outside of the cycle in the original system are invariant in the reconstruction. The bottom center panel displays an observation from a single trial. The right panel shows the R^2_k comparison across models.

ELBO Convergence with Shared Evolution We explore the effect of sharing the evolution parameters between the proposal and target distribution by examining the convergence of the ELBO. Fig. 2 displays the ELBO convergence across epochs as we increase the number of particles (K).

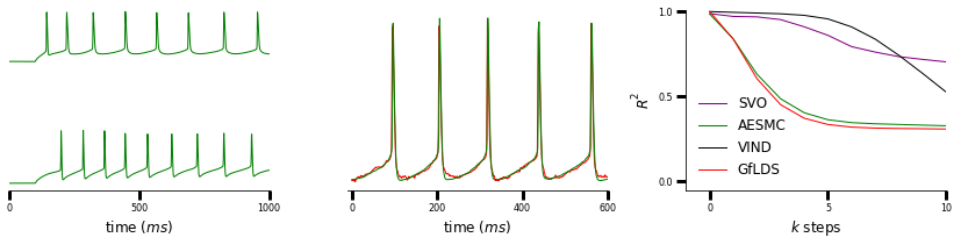

Figure 4: Summary of the Allen results: (left) two trials from the dataset; (center) the data against the predicted observation value using the dynamics learned over a rolling window ten steps ahead. Hyperpolarization and depolarization nonlinearities are predicted by the inferred dynamics; (right) R_k^2 for various models.

The left panel illustrates the ELBO convergence sharing parameters $\varphi = \{\psi, \Sigma\}$ between proposal and target. Larger values of K result in a faster convergence and lower stochastic gradient noise. The center panel displays the ELBO convergence for separate evolution networks for the proposal and the target. Note that for the same value of K, separate evolution networks require a larger number of epochs to converge and the ELBO obtains a lower value with larger stochastic gradient noise. The right panel displays both shared and independent evolution terms for $K = 16$ particles highlighting the difference between the two parameterizations.

4.2 Lorenz Attractor

The Lorenz attractor is a chaotic nonlinear dynamical system defined with 3 independent variables.

$$\begin{aligned} \dot{z}_1 &= \sigma(z_2 - z_1)\,, \\ \dot{z}_2 &= z_1(\rho - z_3) - z_2\,, \\ \dot{z}_3 &= z_1 z_2 - \beta z_3\,. \end{aligned} \tag{39}$$

The system of equations is integrated over 250 time points using $\sigma = 10, \rho = 28, \beta = 8/3$ by generating randomized initial states in $[-10, 10]^3$. Ten dimensional Gaussian observations were produced with a mean specified by a $\mathbf{z}$-dependent neural network. The final dataset consists of 100 trials with 66 for training, 17 for validation and 17 for testing.

The results of the Lorenz experiment are shown in Fig. 3. The inferred latent paths are put together in the left panel illustrating the two cycles. The center panel displays how $R^2_{k=10}$ scales as the latent dimension D_z increases. The R_k^2 stops improving once the correct number of independent variables in the system is reached. The right panel displays the R_k^2 comparison with $D_z = 3$.

4.3 Single Cell Electrophysiology Data

Electrophysiology data was downloaded from the Allen Brain Atlas (Jones et al., 2009). Intracellular voltage recordings were collected from primary Visual Cortex of the mouse, area layer 4. The dataset consists of 40 trials from 5 different cells. The input current in the experiments is a step-function with an amplitude between 80 and 151pA. There were 30 trials reserved for training and 10 for validation. Each trial was divided into five parts and down-sampled from 10,000 time bins to 1,000 in equal-time intervals. Trials were normalized dividing each by its maximal value.

Fig. 4 displays the result of the Allen experiment. The left panel illustrates two trials of 1D observations from the training set. We find that three latent variables provides the best fit. The center panel displays the predicted observation value using the dynamics learned over a rolling window ten steps ahead. Hyperpolarization and depolarization nonlinearities are accurately reconstructed by forward-interpolating the inferred dynamics. The right panel displays the R_k^2 comparison with $D_z = 3$.

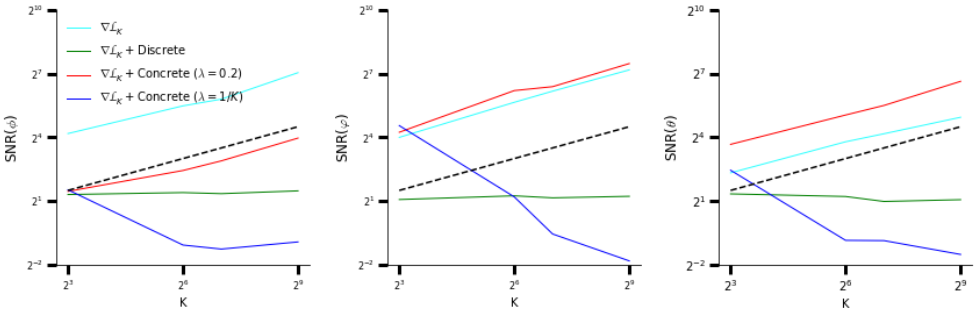

Figure 5: Convergence of SNRs of gradient estimators in the encoder network (left), transition network (center) and decoder network (right) with increasing K. Distinct lines correspond to gradient estimators. The black dashed line with slope 1 illustrates a signal-to-noise-ratio of convergence rate $\mathcal{O}(\sqrt{K})$.

4.4 Signal-to-noise ratio of gradient estimators

For a gradient estimator Δ_K constructed by K particles, the signal-to-noise ratio (SNR) is defined:

$$\text{SNR}_K = \left| \frac{\mathbb{E}[\Delta_K]}{\sqrt{\text{Var}[\Delta_K]}} \right|. \tag{40}$$

In this section, we focus on SNRs of gradient estimators in the encoder network (ϕ), transition network (φ) and decoder network (θ), where the gradient in SNR is taken with respect to ϕ, φ and θ correspondingly.

Fig. 5 presents the empirical results for SNRs of four types of gradient estimators. The expectation and variance in the SNR above are calculated using $N = 100$ gradient samples, which are collected in the middle training stage of running SVO on Fitzhugh-Nagumo data. The 2-norm is used to compute a scalar quantity to define the SNR. We find that the three panels share similar patterns: (1) The gradient estimator without resampling and the relaxed one with temperature $\lambda = 0.2$ possess a signal-to-noise ratio of convergence rate $\mathcal{O}(\sqrt{K})$, which by further investigation is a result of a constant empirical mean and a $\mathcal{O}(K)$ empirical variance. In the IWAE setting, the gradient estimator shrinks to 0 as K increases. In SVO, however, ignoring or relaxing the gradient from the resampling procedure adds a bias to the gradient estimator, hence producing a positive effect of K on SNR; (2) The unbiased CATEGORIAL resampling gradient and relaxed gradient with small bias ($\lambda = (K-1)^{-1}$) suffer from large variance which increases at a rate of $\mathcal{O}(\sqrt{K})$, leading to a relatively low and even decreasing SNR for increasing K.

5 Conclusion

We have sketched a method to construct Smoothed Variational Objectives (SVOs) using Sequential Monte Carlo to perform both inference and parameter estimation in nonlinear dynamical systems. By sharing parameters of the evolution term between the proposal and target, SVO forgoes exchanging expressiveness for tractability. We find that the use of a biased gradient estimate can help address the issue of a degraded signal-to-noise ratio. Using the SVO encoding architecture improves the computational complexity with a cost independent of the number of particles, and the prediction performance, quantified by the k-step reconstruction error across dimensionality reduction and expansion tasks. Future work includes parameterizing an input term to model non-autonomous dynamical systems.

References

Evan Archer, Il Memming Park, Lars Buesing, John Cunningham, and Liam Paninski. Black box variational inference for state space models, 2015.

Jean Bérard, Pierre Del Moral, Arnaud Doucet, et al. A lognormal central limit theorem for particle approximations of normalizing constants. *Electronic Journal of Probability*, 19, 2014.

Yuri Burda, Roger B. Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *CoRR*, abs/1509.00519, 2015.

Daniel Hernandez Diaz, Antonio Khalil Moretti, Ziqiang Wei, Shreya Saxena, John Cunningham, and Liam Paninski. A novel variational family for hidden nonlinear markov models, 2019. URL https://openreview.net/forum?id=SJMO2iCct7.

Yuanjun Gao, Evan Archer, Liam Paninski, and John P. Cunningham. Linear dynamical neural population models through nonlinear embedding. *NIPS 2016*, 2016.

Shixiang Gu, Zoubin Ghahramani, and Richard E. Turner. Neural adaptive sequential monte carlo. *CoRR*, abs/1506.03338, 2015.

Daniel Hernandez, Liam Paninski, and John Cunningham. Variational inference for nonlinear dynamics. *TSW, NIPS 2017*, 2017.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Allan R. Jones, Caroline C. Overly, and Susan M. Sunkin. The allen brain atlas: 5 years and beyond. *Nature Reviews Neuroscience*, 10:821 EP –, 10 2009. URL http://dx.doi.org/10.1038/nrn2722.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013. URL http://arxiv.org/abs/1312.6114.

Genshiro Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5(1):1–25, 1996.

Mike Klaas, Mark Briers, Nando De Freitas, Arnaud Doucet, Simon Maskell, and Dustin Lang. Fast particle smoothing: If i had a million particles. In *Proceedings of the 23rd international conference on Machine learning*, pp. 481–488. ACM, 2006.

Rahul G. Krishnan, Uri Shalit, and David Sontag. Deep kalman filters, 2015.

Rahul G. Krishnan, Uri Shalit, and David Sontag. Structured inference networks for nonlinear state space models, 2016.

Tuan Anh Le, Maximilian Igl, Tom Rainforth, Tom Jin, and Frank Wood. Auto-encoding sequential monte carlo. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJ8c3f-0b.

Scott W. Linderman, Andrew C. Miller, Ryan P. Adams, David M. Blei, Liam Paninski, and Matthew J. Johnson. Recurrent switching linear dynamical systems, 2016.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh. Filtering variational objectives. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 6573–6583. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7235-filtering-variational-objectives.pdf.

Christian Naesseth, Scott Linderman, Rajesh Ranganath, and David Blei. Variational sequential monte carlo. In Amos Storkey and Fernando Perez-Cruz (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 968–977, Playa Blanca, Lanzarote, Canary Islands, 09–11 Apr 2018a. PMLR.

Christian A. Naesseth, Scott W. Linderman, Rajesh Ranganath, and David M. Blei. Variational sequential monte carlo. In *AISTATS*, volume 84 of *Proceedings of Machine Learning Research*, pp. 968–977. PMLR, 2018b.

Chethan Pandarinath, Daniel J. O'Shea, Jasmine Collins, Rafal Jozefowicz, Sergey D. Stavisky, Jonathan C. Kao, Eric M. Trautmann, Matthew T. Kaufman, Stephen I. Ryu, Leigh R. Hochberg, Jaimie M. Henderson, Krishna V. Shenoy, Larry F. Abbott, and David Sussillo. Inferring single-trial neural population dynamics using sequential auto-encoders, 2017.

Liam Paninski and John Cunningham. Neural data science: accelerating the experiment-analysis-theory cycle in large-scale neuroscience. *bioRxiv*, 2017. doi: 10.1101/196949. URL https://www.biorxiv.org/content/early/2017/10/02/196949.

Tom Rainforth, Adam R Kosiorek, Tuan Anh Le, Chris J Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. *arXiv preprint arXiv:1802.04537*, 2018.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Eric P. Xing and Tony Jebara (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR.

Tim Salimans, Diederik P. Kingma, and Max Welling. Markov chain monte carlo and variational inference: Bridging the gap. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pp. 1218–1226, 2015. URL http://jmlr.org/proceedings/papers/v37/salimans15.html.

David Sussillo, Rafal Jozefowicz, L. F. Abbott, and Chethan Pandarinath. Lfads - latent factor analysis via dynamical systems, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou (eds.), *Advances in Neural Information Processing Systems 21*, pp. 1881–1888. Curran Associates, Inc., 2009.