BLACK BOX VARIATIONAL INFERENCE FOR STATE SPACE MODELS

Evan Archer
Department of Statistics and Grossman Center
Columbia University
New York City, NY, United States
evan@stat.columbia.edu

Il Memming Park
Department of Neurobiology and Behavior
Stony Brook University
Stony Brook, NY, United States
memming.park@stonybrook.edu

Lars Buesing*
Department of Statistics and Grossman Center
Columbia University
New York City, NY, United States
lbuesing@google.com

John Cunningham
Department of Statistics and Grossman Center
Columbia University
New York City, NY, United States
jpc2181@columbia.edu

Liam Paninski
Department of Statistics and Grossman Center
Columbia University
New York City, NY, United States
liam@stat.columbia.edu

ABSTRACT

Latent variable time-series models are among the most heavily used tools from machine learning and applied statistics. These models have the advantage of learning latent structure both from noisy observations and from the temporal ordering in the data, where it is assumed that meaningful correlation structure exists across time. A few highly-structured models, such as the linear dynamical system with linear-Gaussian observations, have closed-form inference procedures (e.g. the Kalman Filter), but this case is an exception to the general rule that exact posterior inference in more complex generative models is intractable. Consequently, much work in time-series modeling focuses on approximate inference procedures for one particular class of models. Here, we extend recent developments in stochastic variational inference to develop a 'black-box' approximate inference technique for latent variable models with latent dynamical structure. We propose a structured Gaussian variational approximate posterior that carries the same intuition as the standard Kalman filter-smoother but, importantly, permits us to use the same inference approach to approximate the posterior of much more general, nonlinear latent variable generative models. We show that our approach recovers accurate estimates in the case of basic models with closed-form posteriors, and more interestingly performs well in comparison to variational approaches that were designed in a bespoke fashion for specific non-conjugate models.

1 INTRODUCTION

Latent variable models are commonplace in time-series analysis, with applications across statistics, engineering, the sciences, finance and economics. The core approach is to assume that latent variables $\mathbf{z}_1, \ldots, \mathbf{z}_T \in \mathbb{R}^n$, which are correlated across t, underlie correlated observations $\mathbf{x}_1, \ldots, \mathbf{x}_T \in \mathbb{R}^m$. Standard models for latent dynamics include the linear dynamical system (LDS) and hidden Markov models. While each approach comes with a distinct model and set of computational tools, often the basic goal of inference is the same: to discern the filtering distribution

*Current affiliation: Google DeepMind, London, UK.

$p(\mathbf{z}_t|\mathbf{x}_{1:t})$ and the smoothing distribution $p(\mathbf{z}_t|\mathbf{x}_{1:T})$ of the latent variables. Closed-form expressions for these distributions are available when the overall probabilistic model has tree or chain structure that admits closed-form message passing. In general, inference in non-Gaussian or nonlinear models requires numerical approximation or sampling.

Markov chain Monte Carlo sampling and particle filtering for general time-series models are well-developed but typically do not scale well to large-scale problems. Even when a model $p_\theta(\mathbf{x}, \mathbf{z})$, with parameters θ, has been trained, we can only access an analytically-intractable posterior through further sampling. Here, we take a variational approach to time-series modeling: rather than attempting to compute the posterior $p_\theta(\mathbf{z}|\mathbf{x})$ of our generative model, we approximate it with a distribution $q_\phi(\mathbf{z}|\mathbf{x})$ with variational parameters ϕ. Inference proceeds by simultaneously optimizing p_θ (through its model parameters θ) and q_ϕ (through its variational parameters ϕ) such that q_ϕ approximates the true posterior.

Our main contributions are (1) a structured approximate posterior that can express temporal dependencies, and (2) a fast and scalable inference algorithm. We propose a multivariate Gaussian approximate posterior with block tri-diagonal inverse covariance, and formulate an algorithm that scales (in both time and space complexity) only linearly in the length of the time-series. For inference, we make use of recent advances in variational inference, stochastic gradient variational Bayes (SGVB) (Rezende et al., 2014; Kingma & Welling, 2013; Kingma et al., 2014), to learn an approximate posterior with a complex functional dependence upon the observations $\mathbf{x}$. Using this approach we are able to learn a neural network (NN) that maps $\mathbf{x}$ into the smoothed posterior $q(\mathbf{z}|\mathbf{x})$ (sometimes called the recognition model). This approach is 'black-box' in the sense that the inference algorithm does not depend explicitly upon the functional form of the generative model p_θ.

Our motivations lie in the study of high-dimensional time-series, such as neural spike-train recordings (Kao et al., 2015). We seek to infer trajectories $\mathbf{z}$ that provide insight into the latent, low-dimensional structure in the dynamics of such data. Recent, related approaches to variational inference in time-series models focus upon the design and learning of rich generative models capable of capturing the statistical structure of large, complex datasets (Gan et al., 2015; Chung et al., 2015; Bayer & Osendorfer, 2014). In contrast, our focus is upon computationally efficient inference in structured, interpretable parameterizations that build upon methods fundamental in scientific applications.

We apply our smoothing approach for approximate posterior inference of a well-studied generative model: the Poisson linear dynamical system (PLDS) model. We find that our general, black-box approach outperforms a specialized variational Bayes expectation maximization (VBEM) approach (Emtiyaz Khan et al., 2013) to inference in PLDS, reaching comparable solutions to VBEM before it can complete a single EM iteration. Additionally, we apply our method to inference in a one-dimensional, nonlinear dynamical system, showing that we are able to accurately recover nonlinear relationships in the posterior mean.

2 Stochastic gradient variational Bayes

In variational inference we approximate an intractable posterior distribution $p_\theta(\mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{x}, \mathbf{z})/p_\theta(\mathbf{x})$ with $q_\phi(\mathbf{z}|\mathbf{x})$[1] that comes from a tractable class (e.g., the Gaussian family) and is parameterized by variational parameters ϕ. We learn ϕ and θ together by optimizing the *evidence lower bound* (ELBO) of the marginal likelihood (Jordan et al., 1999), given by,

$$\log p_\theta(\mathbf{x}) \geq \mathcal{L}(\theta, \phi; \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[-\log q_\phi(\mathbf{z}|\mathbf{x}) + \log p_\theta(\mathbf{x}, \mathbf{z})\right] \tag{1}$$

$$= H(q_\phi(\mathbf{z}|\mathbf{x})) + \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}, \mathbf{z})\right]. \tag{2}$$

The quantity $\mathcal{L}(\theta, \phi; \mathbf{x})$ is the ELBO, and $H(q_\phi(\mathbf{z}|\mathbf{x}))$ is the entropy of the approximating posterior. Our goal is to differentiate $\mathcal{L}(\theta, \phi)$ with respect to ϕ and θ so as to maximize $\mathcal{L}$,

$$\nabla\mathcal{L}(\theta, \phi; \mathbf{x}) := \nabla\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\underbrace{-\log q_\phi(\mathbf{z}|\mathbf{x}) + \log p_\theta(\mathbf{x}, \mathbf{z})}_{:=f_{\{\theta,\phi\}}(\mathbf{z})}]. \tag{3}$$

[1] In many approaches to variational inference the dependence of upon $\mathbf{x}$ is dropped; in our case, the parameterization of $q_\phi(\mathbf{z}|\mathbf{x})$ may depend explicitly upon the observations $\mathbf{x}$.

For the remainder of this section, we use the notation $f_{\{\theta,\phi\}}(\mathbf{z}) = -\log q_\phi(\mathbf{z}|\mathbf{x}) + \log p_\theta(\mathbf{x},\mathbf{z})$. Typically at least some terms of eq. 3 cannot be integrated in closed form. While it is often possible to estimate the gradient by sampling directly from $q(\mathbf{z}|\mathbf{x})$, in general the approximate gradient exhibits high variance (Paisley et al., 2012). One approach to addressing this difficulty, independently proposed by Kingma & Welling (2013), Rezende et al. (2014) and Titsias & Lázaro-Gredilla (2014), is to compute the integral using the "reparameterization trick": choose an easy-to-sample random variable ϵ with distribution $p(\epsilon)$ and parameterize $\mathbf{z}$ through a function g of observations $\mathbf{x}$ and parameters ϕ,

$$\mathbf{z} = g_\phi(\mathbf{x}, \epsilon). \tag{4}$$

The point of this notation is to make clear that $g_\phi(\mathbf{x}, \cdot)$ is a *deterministic* function: all randomness in q comes from the random variable ϵ. This allows us to approximate the gradient using the simple estimator,

$$\nabla \mathbb{E}_{q_\phi(\mathbf{z})}\left[f_{\{\theta,\phi\}}(\mathbf{z})\right] = \mathbb{E}_{p(\epsilon)}\left[\nabla f_{\{\theta,\phi\}}(g_\phi(\mathbf{x},\epsilon))\right] \approx \frac{1}{L}\sum_{l=1}^{L} \nabla f_{\{\theta,\phi\}}(g_\phi(\mathbf{x},\epsilon^l)), \tag{5}$$

where ϵ^l are iid samples from $p(\epsilon)$. In Kingma & Welling (2013) this estimator is referred to as the Stochastic Gradient Variational Bayes (SGVB) estimator. Empirically, eq. 5 has much lower variance than previous sampling-based approaches to the estimation of eq. 3 (Kingma & Welling, 2013; Titsias & Lázaro-Gredilla, 2014).

An important property of eq. 5 is that it does not depend upon the particular form of $f_{\{\theta,\phi\}}(\cdot)$: we need only be able to evaluate it at the samples ϵ^i. It is in this sense that our approach is "black-box": in principle, inference works the same way regardless of our choice of generative model p_θ. In practice, of course, different modeling choices will affect the computation time and the convergence rate of the method.

The estimator also permits significant freedom in our parameterization of the transformation $g_\phi(\mathbf{x}, \cdot)$: for inference, we just need to be able to differentiate g with respect to ϕ. While it is possible to use the SGVB approach with a separate set of parameters ϕ_t for each observation (as in Hoffman et al. (2013), for instance), much recent work has used deep neural networks (DNNs) to train a function that maps directly into the posterior (Rezende et al., 2014; Kingma & Welling, 2013; Kingma et al., 2014). Under this approach, with a trained $g_\phi(\mathbf{x}, \cdot)$, no additional gradient steps are needed to obtain $q(\mathbf{z}|\mathbf{x})$ for new observations $\mathbf{x}$.

3 VARIATIONAL APPROACH TO STATE-SPACE MODELING

Using a black-box inference approach, learning a state-space model is in part just a matter of parameterizing a generative model p_θ with time-series structure. However, the posterior $p(\mathbf{z}|\mathbf{x})$ will in general have temporal correlation structure inadequately captured by the approximate posteriors studied in most previous variational inference literature. The first challenge, then, is to formulate an approximate posterior expressive enough to capture the temporal correlations characteristic of time-series models.

In the timeseries setting, we take $\mathbf{x}$ and $\mathbf{z}$ as "stacked" versions of the observation and latent state, respectively, at a particular time t. In symbols: we let $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_t, \ldots, \mathbf{x}_T)$ and $\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_t, \ldots, \mathbf{z}_T)$, where $\mathbf{x}_t \in \mathbb{R}^m$ and $\mathbf{z}_t \in \mathbb{R}^n$.

3.1 GAUSSIAN APPROXIMATE POSTERIOR

One common, convenient choice of approximate posterior is the multivariate normal. In the notation of Section 2, the multivariate normal comes about if we choose $\epsilon \sim \mathcal{N}(0, I)$ and take $g_\phi(\mathbf{x}, \cdot)$ to be an affine function (Titsias & Lázaro-Gredilla, 2014). We can then express a sample $\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})$ as,

$$\mathbf{z} = g_\phi(\mathbf{x}, \epsilon) \tag{6}$$

$$= \mu_\phi(\mathbf{x}) + R_\phi(\mathbf{x})\epsilon, \tag{7}$$

so that $\mathbf{z}$ is distributed as multivariate normal with mean $\mu_\phi(\mathbf{x})$ and covariance $\Sigma_\phi(\mathbf{x}) = R_\phi(\mathbf{x})R_\phi(\mathbf{x})^{\mathrm{T}}$:

$$q(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mu_\phi(\mathbf{x}), \Sigma_\phi(\mathbf{x})\right). \tag{8}$$

It is easy to sample from a Gaussian approximate posterior using eq. 6, and the entropy term within eq. 2 has a closed form:

$$H(q_\phi(\mathbf{z}|\mathbf{x})) = -\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log q_\phi(\mathbf{z}|\mathbf{x})\right] = \frac{nT}{2}\left(1 + \log(2\pi)\right) + \frac{1}{2}\log\det(\Sigma_\phi(\mathbf{x})). \tag{9}$$

A potential downside of the Gaussian approach that Σ is $nT \times nT$, and so the number of parameters scales quadratically in T. This makes it difficult to manage and learn for large-scale datasets. A simple workaround is to consider Σ with a special structure that reduces the effective number of parameters. Possible examples include using diagonal covariance (fully-factorized, or "mean field" approximation) (Bishop, 2006), or a diagonal covariance matrix plus low-rank matrix (for instance, a sum of outer products) (Rezende et al., 2014).

3.2 Smoothing Gaussian approximate posterior

For modeling time-series data, we seek an approximate posterior capable of expressing our strong expectation that the latent variables change smoothly over time. While the Gaussian approximate posterior of eq. 8 can represent arbitrary correlation structure, we propose a Gaussian approximate posterior whose parameterization scales only linearly in T. To do so, we borrow from the toolkit of the standard Kalman filter. In an LDS model with Gaussian observations, the posterior is a multivariate Gaussian with a block tri-diagonal inverse covariance. This block-tridiagonal structure results from (and expresses) the conditional independence properties of the LDS prior.

To enable our approximate posterior to express the same correlation structure we parameterize the inverse covariance of eq. 8, Σ^{-1}, to be block tri-diagonal. Our final posterior takes the form:

$$q(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mu_\phi(\mathbf{x}), \left[R_\phi(\mathbf{x})R_\phi(\mathbf{x})^{\mathrm{T}}\right]^{-1}\right), \tag{10}$$

where $\mu_\phi(\mathbf{x})$ is the posterior mean and $R_\phi(\mathbf{x})$ is a lower block bi-diagonal matrix with $n \times n$ blocks [2].

3.2.1 Computation

In this subsection we drop subscripts and functional notation for clarity and refer, for instance, to $\Sigma_\phi(\mathbf{x})$ as Σ. We can perform inference efficiently by exploiting the special structure of Σ.

While Σ is in general a dense matrix, we parameterize Σ^{-1} as block tri-diagonal. Since Σ is symmetric, in practice we represent only the diagonal and first block off-diagonal matrices of Σ^{-1}. Matrix inversion and sampling may be performed quickly using the Cholesky decomposition[3], $\Sigma^{-1} = RR^{\mathrm{T}}$. The computation of the lower-triangular Cholesky factor, R, is linear (in both time in space) in the length of the time-series T.

We can sample as in eq. 6, where now:

$$\mathbf{z} = \mu + R^{-\mathrm{T}}\epsilon. \tag{11}$$

For an arbitrary matrix $R \in \mathbb{R}^{nT \times nT}$, computation of $R^{-\mathrm{T}}\epsilon$ scales cubically in the dimensionality of the matrix. However, by exploiting the lower-triangular structure of R, matrix inversion scales only linearly (Trefethen & Bau III, 1997). The entropy of q is also easy to compute since $\log\det(\Sigma) = -2\log\det(R) = -2\sum_{i=1}^{T}\log(R_{ii})$.

In short, for learning ϕ and θ we need never explicitly represent any part of Σ. For data analysis and model comparison, however, it may be useful to compute the covariance $\mathrm{cov}(\mathbf{z}_t, \mathbf{z}_{t+1})$. These

[2] A lower block bi-diagonal matrix has only non-zero diagonal and (first) lower-diagonal blocks.

[3] Block structure is frequently exploited in computation of the Cholesky decomposition; see for instance Björck (1996).

covariances correspond to the block-diagonal and first block off-diagonals of Σ, and may also be computed efficiently (Jain et al., 2007).

Our overall approach is closely related to the standard forward-backward algorithm used for instance in Kalman smoothing. However, there is a major technical distinction between its standard use (e.g., in expectation maximization) and our approach: we explicitly differentiate parameters ϕ through the matrix factorization $\Sigma^{-1} = RR^{\mathrm{T}}$.

4 PARAMETERIZATION OF THE SMOOTHING POSTERIOR

While eq. 10 succinctly states the general mathematical form of the smoothing posterior, the practical performance of the algorithm depends upon the specifics of the parameterization. There are many possible parameterizations, especially since the parameters $\Sigma_\phi(\mathbf{x})$ and $\mu_\phi(\mathbf{x})$ may be arbitrary functions of observations $\mathbf{x}$. To illustrate, we discuss two distinct parameterizations. We use the notation $P = \mathrm{NN}_{\phi_P}(\mathbf{x})$ to indicate that parameter P is defined as a function of inputs $\mathbf{x}$ through a neural network $\mathrm{NN}_{\phi_P}(\cdot)$ with parameters ϕ_P. The parameters of all networks are incorporated into ϕ:

$$\phi = \{\phi_{P_1}, \phi_{P_2}, \phi_{P_3}, \dots\}. \tag{12}$$

4.1 DIAGONAL AND BLOCK OFF-DIAGONAL PARAMETERIZATION

We can naturally parameterize $\mu_\phi(\mathbf{x})$ and $\Sigma_\phi(\mathbf{x})$ of eq. 10 using 3 neural networks. We use one neural network to represent a map $\mathbf{x}_t \to \mu_t$,

$$\mu_t = \mathrm{NN}_{\phi_\mu}(\mathbf{x}_t), \tag{13}$$

where μ_t is a $n \times 1$ segment of μ, and $\mu = (\mu_1, \mu_2, \dots, \mu_T)$. We can parameterize the block tri-diagonal covariance $\Sigma_\phi(\mathbf{x})^{-1}$,

$$\Sigma_\phi(\mathbf{x})^{-1} = \begin{bmatrix} D_0 & B_0^{\mathrm{T}} & & \\ B_0 & D_1 & B_1^{\mathrm{T}} & \\ & \ddots & \ddots & B_{T-1}^{\mathrm{T}} \\ & & B_{T-1} & D_T \end{bmatrix}, \tag{14}$$

by parmeterizing each of the blocks separately:

$$D_t = \mathrm{NN}_{\phi_D}(\mathbf{x}_t) \tag{15}$$

$$B_t = \mathrm{NN}_{\phi_B}(\mathbf{x}_t, \mathbf{x}_{t-1}). \tag{16}$$

In practice, we found it necessary to enforce the positive-definiteness of the covariance by adding a diagonal matrix αI to $\Sigma_\phi(\mathbf{x})^{-1}$, where $\alpha > 0$ is a fixed constant. In the experiments, we refer to this parameterization as VILDSblk.

4.2 PRODUCT-OF-GAUSSIANS APPROXIMATE POSTERIOR

We can also define the approximate posterior through a product of Gaussian factors, $q(\mathbf{z}|\mathbf{x}) \propto r_1(\mathbf{z}|\mathbf{x}) r_0(\mathbf{z})$, where:

$$r_0(\mathbf{z}) := \mathcal{N}(\mathbf{z}|0, \mathbf{D}) \tag{17}$$

$$r_1(\mathbf{z}|\mathbf{x}) := \mathcal{N}(\mathbf{z}|\mathbf{M}_\phi(\mathbf{x}), \mathbf{C}_\phi(\mathbf{x})), \tag{18}$$

$\mathbf{D}$ and $\mathbf{C}$ are $nT \times nT$ matrices and $\mathbf{M}$ is a nT-dimensional vector. In this set-up, we can view r_0 as a prior. In terms of eq. 10, the final posterior is then given by:

$$\Sigma_\phi(\mathbf{x}) = \left(\mathbf{D}^{-1} + \mathbf{C}_\phi^{-1}(\mathbf{x})\right)^{-1} \tag{19}$$

$$\mu_\phi(\mathbf{x}) = \Sigma_\phi(\mathbf{x}) \mathbf{C}_\phi^{-1}(\mathbf{x}) \mathbf{M}_\phi(\mathbf{x}). \tag{20}$$

In order to be a parameterization of the smoothing posterior, eq. 10, $\mathbf{D}^{-1}$ and $\mathbf{C}^{-1}$ must be block tri-diagonal. The multiplicative interaction between the posterior mean and covariance leads to

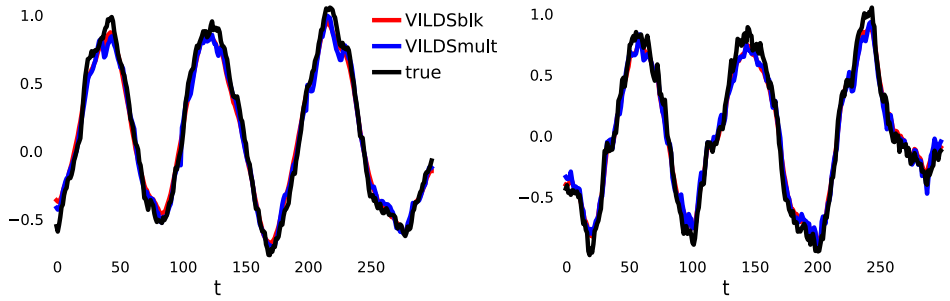

Figure 1: Posterior mean inference compared with ground truth. Each panel shows the posterior mean along a dimension of the two-dimensional ($n = 2$) state space of a Kalman filter. We show 300 time-points of the posterior means from a $T = 5000$ sample Kalman filter experiment. We fit using VILDS with the parameterization described in Section 4.1 (VILDSblk) and that described in Section 4.2 (VILDSmult). The true posterior means computed using the closed-form Kalman filter equations **(black)** agree closely with those recovered using VILDSmult **(blue)** and VILDSblk **(red)**.

different performance from the parameterization described in Section 4.1. Further, we can choose $\mathbf{D}$ to initialize the means with a given degree of smoothness, which is not possible in the formulation of Section 4.1. In the experiments, we refer to this parameterization as VILDSmult; in Appendix A we describe the specific parameterization we used for $\mathbf{C}^{-1}$ and $\mathbf{D}^{-1}$.

5 Experiments

In the experiments, we refer to the SGVB with the smoothing approximate posterior as VILDS. We refer to SGVB with an approximate posterior independent across time as mean field (MF). The mean field posterior is given by,

$$q(\mathbf{z}|\mathbf{x}) = \prod_{t=1}^{T} \mathcal{N}(\mathbf{z}_t; \mu_t, V_t), \tag{21}$$

where V_t is a full $n \times n$ covariance matrix. We optimize all parameters by gradient ascent, using the SGVB approach with $L = 1$ to estimate the gradient with eq. 5.

For training VILDS and MF, we performed gradient descent on all parameters $\{\theta, \phi\}$ of the generative model and approximate posterior. We tried several adaptive gradient stochastic optimization methods, including: ADAM (Kingma & Ba, 2014), Adadelta (Zeiler, 2012), Adagrad (Duchi et al., 2011) and RMSprop (Tieleman & Hinton, 2012). In the experiments we show here, we used Adadelta to learn all parameters. We gradually decreased the base learning rate by a factor of 10 after a period of 20 "epochs" without an increase of the objective function.

5.1 Kalman filter model

First, we illustrate the efficacy of our approach by showing that we can recover the analytic posterior of a Kalman filter model. Under a Kalman filter model, the latents are governed by an LDS,

$$\mathbf{z}_t = \mathbf{A}\mathbf{z}_{t-1} + \boldsymbol{\epsilon}_t, \tag{22}$$

with Gaussian innovation noise with covariance matrix $\mathbf{Q}$, $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{Q})$. Observations are coupled to the latents through a loading matrix $\mathbf{C}$,

$$\mathbf{x}_t = \mathbf{C}\mathbf{z}_t + \eta_t, \tag{23}$$

and η_t are Gaussian noise with diagonal covariance.

We simulated 5000 time-points from a 2-dimensional latent dynamical system model, with 100-dimensional linear observations. We parameterize the VILDS approximate posterior using a 5-layer, dense NN for each of $\mu_\phi(\mathbf{x})$ and $R_\phi(\mathbf{x})$. We use a rectified-linear nonlinearity between each layer, followed by a linear output layer mapping into the parameterization. We compare both of the approximate posterior parameterizations described in Section 4. We refer to the parameterization of

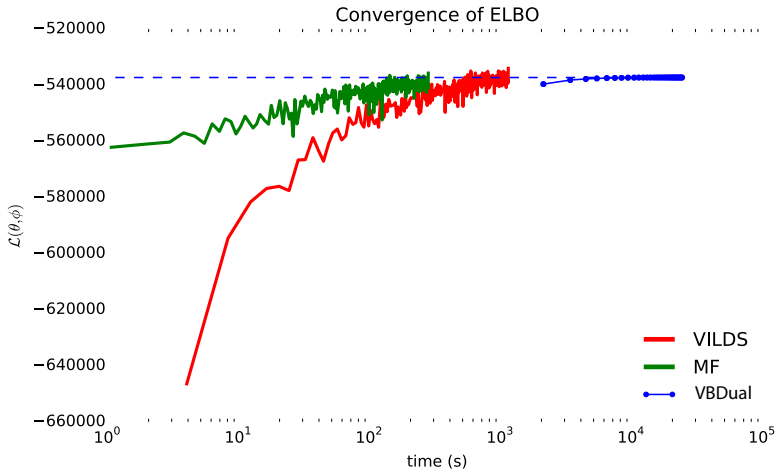

Figure 2: Speed comparison: ELBO convergence vs time (seconds) for VILDS, MF and VBDual. VBDual is fit by an iterative procedure that optimizes a dual-space cost function for each E-step (Emtiyaz Khan et al., 2013). The first VBDual E-step takes many iterations to converge, and causes the long gap from 0 seconds to the first VBDual datapoint. Subsequent VBDual E-steps are less time-consuming. VILDS and MF are learned by stochastic gradient descent using the adaptive-gradient technique, Adadelta (Zeiler, 2012). Gradients are computed on minibatches of size 100, and each datapoint is collected after 100 minibatches (one "epoch"). Both MF and VILDS were run for 500 epochs. VILDS achieves the highest ELBO value, followed by MF and then VBDual. VILDS achieves ELBO values comparable to VBDual before VBDual can complete a single EM iteration.

Section 4.1 as VILDSblk, and that of Section 4.2 as VILDSmult. For both choices of approximate posterior, the VILDS smoothed posterior means (Fig. 1) show good agreement with the true Kalman filter posterior. The VILDS smoothed posterior variances, $\mathbb{V}[\mathbf{z}_t]$ and $\mathrm{cov}(\mathbf{z}_t, \mathbf{z}_{t-1})$ also show good agreement with the Kalman filter posterior covariance (not shown).

5.2 Poisson LDS (PLDS)

The Kalman filter/smoother is exact for an LDS with linear-Gaussian observations. A common generalization in the literature is an LDS with non-Gaussian observations. One well-studied example is the Poisson LDS (PLDS). Under this model, the latents are again governed by an LDS,

$$\mathbf{z}_t = \mathbf{A}\mathbf{z}_{t-1} + \boldsymbol{\epsilon}_t, \tag{24}$$

with Gaussian innovation noise with covariance matrix $\mathbf{Q}$, $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{Q})$. Observations are modulated by a log-rate $\mathbf{r}_t$, which is coupled to the latent state $\mathbf{z}_t$ via a loading matrix $\mathbf{C}$,

$$\mathbf{r}_t = \mathbf{C}\mathbf{z}_t + \mathbf{d}. \tag{25}$$

The vector $\mathbf{d}$ is a vector bias term for each element of the observation. Given the log-rate $\mathbf{r}_t$, observations $\mathbf{x}_t \in \mathbb{N}^m$ are Poisson-distributed,

$$x_{k,t}|\mathbf{z}_t \sim \mathrm{Poisson}(\exp(r_{k,t})). \tag{26}$$

With Poisson observations, the posterior does not have a closed form. Several methods have been proposed for approximate learning and inference in the special case of the PLDS (Buesing et al., 2014; 2012; Macke et al., 2011); Laplace approximation is also frequently used (Paninski et al., 2010; Fahrmeir & Kaufmann, 1991). We compare VILDS to the variational Bayes expectation-maximization approach proposed by Emtiyaz Khan et al. (2013). This VBEM approach uses a full, unconstrained Gaussian as a variational approximate posterior q_ϕ, and performs EM iterations through a dual-space parameterization. We refer to it by the abbreviation VBDual, to emphasize this dual-space parameterization. We parameterize both the MF and VILDS approximate posteriors using a 5-layer, dense NN for each of $\mu_\phi(\mathbf{x})$ and $R_\phi(\mathbf{x})$. For VILDS, we use the parameterization

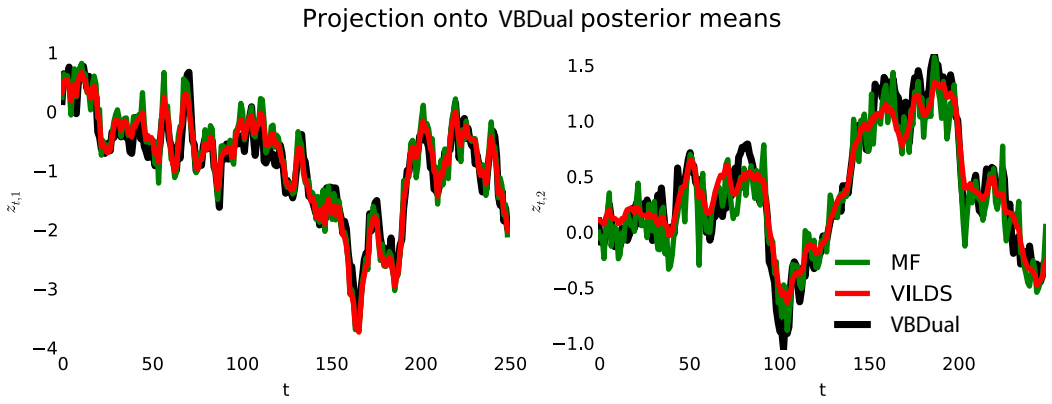

Figure 3: Comparison of posterior means learned using the mean field approximate posterior **(green)**, VILDS **(red)** and VBDual with an unstructured Gaussian approximate posterior **(black)**. The model has a rotational invariance, and so we project the MF and VILDS posteriors onto the VBDual posterior means using least squares. The posterior mean trajectories learned by VILDS are visibly smoother than the MF posterior mean trajectories.

of Section 4.2. We use a rectified-linear nonlinearity between each layer, followed by a linear output layer mapping into the parameterization. We simulated $T = 5000$ samples from a PLDS model with $n = 2$ latent states and $m = 100$ observation dimensions. We initialized all three methods (VILDS, MF and VBDual) using the nuclear-norm minimization methods outlined in Pfau et al. (2013).

To better illustrate the timecourse of learning, each epoch consisted of only 100 minibatches, where each minibatch was of size 100. A single gradient step was taken for each minibatch. We ran both MF and VILDS for a fixed 500 iterations. We find that VILDS reaches a higher ELBO value than either MF or VBDual, and does so before VBDual can complete a single expectation-maximization iteration (see Fig. 2). Further, the posterior means learned by VILDS are smoother than those learned using the MF approximate posterior (see Fig. 3).

5.3 Nonlinear dynamics simulation

VILDS can perform approximate posterior inference for nonlinear-dyanamical generative models. To illustrate, we simulated 5000 samples from a toy one-dimensional nonlinear dynamical model given by:

$$\mathbf{z}_t = -\frac{1}{2}\mathbf{z}_{t-1} + 5\cos(.5\mathbf{z}_{t-1}) + .5\epsilon_t \tag{27}$$

$$\mathbf{x}_t = \frac{1}{2}\mathbf{z}_t + .5\eta_t, \tag{28}$$

where ϵ_t and η_t are each iid $\mathcal{N}(0, 1)$ random variables. For the approximate posterior, we parameterized both the μ and R using 8-layer networks where each layer has only a single unit, and rectified-linear nonlinearity. We use the LDS-inspired parameterization of Section 4.2. As shown in Fig. 4, VILDS is capable of recovering the nonlinear relationship in the state space. For these experiments, we held the generative model parameters θ fixed and learned only ϕ.

6 Conclusion

We proposed a Gaussian variational approximate posterior with block tri-diagonal covariance structure capable of expressing "smoothed" trajectories of a time-series posterior. Exploiting the block tri-diagonal covariance structure, inference scales only linearly (in both time and space complexity) in the length T of a time-series. Using the SGVB approach to variational inference, we can perform approximate inference for a wide class of latent variable generative models.

Despite the generality of the inference algorithm, the approach is limited by the Gaussian approximate posterior: most latent variable time-series generative models have non-Gaussian posteriors.

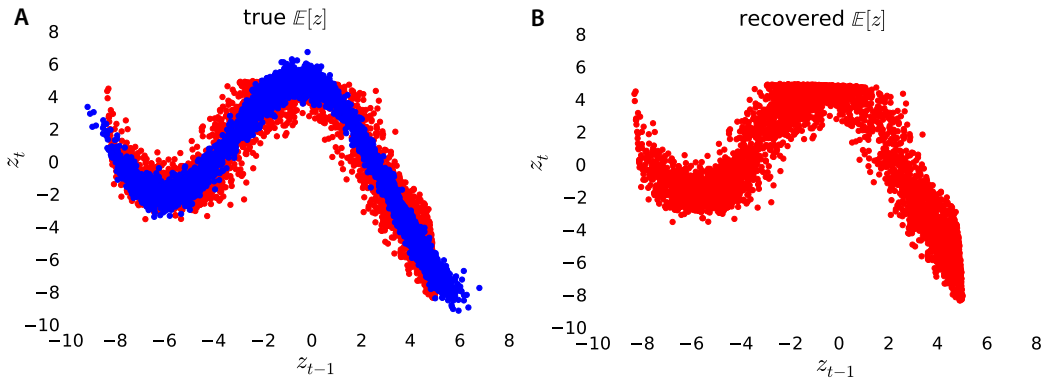

Figure 4: True and VILDS-fit posterior means for nonlinear dynamics simulation. VILDS was fit to 5000 samples drawn from the nonlinear dynamical system described in eq. 27. Each red point represents a single ordered pair $(\mathbb{E}[z_t], \mathbb{E}[z_{t-1}])$, while each blue point represents a "true" latent state $(\mathbf{z}_t, \mathbf{z}_{t-1})$. **(A)** We illustrate the nonlinear dynamical relationship between $\mathbf{z}_t$ and $\mathbf{z}_{t-1}$; in a linear dynamical system this relationship would be a straight line. In red, we show the posterior means recovered by VILDS. **(B)** The VILDS posterior means of **(A)** plotted alone, for comparison.

One possible route forward are the methods of Rezende & Mohamed (2015) and Dinh et al. (2014), which permit learning and inference using a non-Gaussian approximate posterior within the SGVB framework.

We implemented all methods in Python using Theano with the Lasagne library (Bastien et al., 2012; Bergstra et al., 2010). We plan to release the source code on Github shortly.

As we were preparing this manuscript we became aware of Krishnan et al. (2015), which studies a closely-related (but distinct) method. In future work, we will plan to perform detailed comparisons between the methods.

ACKNOWLEDGMENTS

Funding for this research was provided by DARPA N66001-15-C-4032, Google Faculty Research award, and ONR N00014-14-1-0243 (LP); Simons Global Brain Research Award 325171 (LP and JC); Sloan Research Fellowship (JPC). We thank David Carlson and Megan McKinney, Esq. for useful comments on the manuscript, and Gabriel Synnaeve for making his Python code publicly available.

REFERENCES

Bastien, Frédéric, Lamblin, Pascal, Pascanu, Razvan, Bergstra, James, Goodfellow, Ian, Bergeron, Arnaud, Bouchard, Nicolas, Warde-Farley, David, and Bengio, Yoshua. Theano: new features and speed improvements. *arXiv preprint arXiv:1211.5590*, 2012.

Bayer, Justin and Osendorfer, Christian. Learning stochastic recurrent networks. *arXiv preprint arXiv:1411.7610*, 2014.

Bergstra, James, Breuleux, Olivier, Bastien, Frédéric, Lamblin, Pascal, Pascanu, Razvan, Desjardins, Guillaume, Turian, Joseph, Warde-Farley, David, and Bengio, Yoshua. Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for scientific computing conference (SciPy)*, volume 4, pp. 3. Austin, TX, 2010.

Bishop, Christopher M. *Pattern recognition and machine learning*. springer, 2006.

Björck, Ake. *Numerical methods for least squares problems*. Siam, 1996.

Buesing, Lars, Macke, Jakob H, and Sahani, Maneesh. Spectral learning of linear dynamics from generalised-linear observations with application to neural population data. In *Advances in neural information processing systems*, pp. 1682–1690, 2012.

Buesing, Lars, Machado, Timothy A, Cunningham, John P, and Paninski, Liam. Clustered factor analysis of multineuronal spike data. In *Advances in Neural Information Processing Systems*, pp. 3500–3508, 2014.

Chung, Junyoung, Kastner, Kyle, Dinh, Laurent, Goel, Kratarth, Courville, Aaron, and Bengio, Yoshua. A recurrent latent variable model for sequential data. *arXiv preprint arXiv:1506.02216*, 2015.

Dinh, Laurent, Krueger, David, and Bengio, Yoshua. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

Duchi, John, Hazan, Elad, and Singer, Yoram. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

Emtiyaz Khan, Mohammad, Aravkin, Aleksandr, Friedlander, Michael, and Seeger, Matthias. Fast dual variational inference for non-conjugate latent gaussian models. In *Proceedings of The 30th International Conference on Machine Learning*, pp. 951–959, 2013.

Fahrmeir, Ludwig and Kaufmann, Heinz. On Kalman filtering, posterior mode estimation and Fisher scoring in dynamic exponential family regression. *Metrika*, 38(1):37–60, 1991.

Gan, Zhe, Li, Chunyuan, Henao, Ricardo, Carlson, David, and Carin, Lawrence. Deep temporal sigmoid belief networks for sequence modeling. *arXiv preprint arXiv:1509.07087*, 2015.

Hoffman, Matthew D, Blei, David M, Wang, Chong, and Paisley, John. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Jain, Jitesh, Li, Hong, Cauley, Stephen, Koh, Cheng-Kok, and Balakrishnan, Venkataramanan. Numerically stable algorithms for inversion of block tridiagonal and banded matrices. *Purdue ECE Technical Reports. Paper 357.*, 2007.

Jordan, Michael I, Ghahramani, Zoubin, Jaakkola, Tommi S, and Saul, Lawrence K. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

Kao, Jonathan C., Nuyujukian, Paul, Ryu, Stephen I., Churchland, Mark M., Cunningham, John P., and Shenoy, Krishna V. Single-trial dynamics of motor cortex and their applications to brain-machine interfaces. *Nature Communications*, 6:7759+, July 2015. ISSN 2041-1723. doi: 10.1038/ncomms8759.

Kingma, Diederik and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, Diederik P and Welling, Max. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kingma, Diederik P, Mohamed, Shakir, Rezende, Danilo Jimenez, and Welling, Max. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pp. 3581–3589, 2014.

Krishnan, Rahul G., Shalit, Uri, and Sontag, David. Deep Kalman filters. *arXiv preprint arXiv:1511.05121*, 2015.

Macke, Jakob H, Buesing, Lars, Cunningham, John P, Byron, M Yu, Shenoy, Krishna V, and Sahani, Maneesh. Empirical models of spiking in neural populations. In *Advances in neural information processing systems*, pp. 1350–1358, 2011.

Paisley, John, Blei, David, and Jordan, Michael. Variational Bayesian inference with stochastic search. *arXiv preprint arXiv:1206.6430*, 2012.

Paninski, Liam, Ahmadian, Yashar, Ferreira, Daniel Gil, Koyama, Shinsuke, Rad, Kamiar Rahnama, Vidne, Michael, Vogelstein, Joshua, and Wu, Wei. A new look at state-space models for neural data. *Journal of computational neuroscience*, 29(1-2):107–126, 2010.

Pfau, David, Pnevmatikakis, Eftychios A, and Paninski, Liam. Robust learning of low-dimensional dynamics from large neural ensembles. In *Advances in Neural Information Processing Systems*, pp. 2391–2399, 2013.

Rezende, Danilo Jimenez and Mohamed, Shakir. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

Rezende, Danilo Jimenez, Mohamed, Shakir, and Wierstra, Daan. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Tieleman, Tijmen and Hinton, Geoffrey. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4, 2012.

Titsias, Michalis and Lázaro-Gredilla, Miguel. Doubly stochastic variational bayes for non-conjugate inference. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1971–1979, 2014.

Trefethen, Lloyd N and Bau III, David. *Numerical linear algebra*, volume 50. Siam, 1997.

Zeiler, Matthew D. ADADELTA: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

A SMOOTHING APPROXIMATE POSTERIOR WITH EXPLICIT FORWARD/BACKWARD

The posterior mean and covariances may be computed by the standard Kalman forward-backward algorithm. To see this, we can write the posterior in matrix form as the product of two Gaussians. We have

$$r_0(\mathbf{z}) := \mathcal{N}(\mathbf{z}|0, \mathbf{D}) \tag{29}$$
$$r_1(\mathbf{z}|\mathbf{x}) := \mathcal{N}(\mathbf{z}|\mathbf{M}_\phi(\mathbf{x}), \mathbf{C}_\phi(\mathbf{x})), \tag{30}$$

where,

$$\mathbf{Q} = I_{T\times T} \otimes Q, \quad \mathbf{A} = \begin{bmatrix} 0 & & & \\ 1 & 0 & & \\ & \ddots & \ddots & \\ & & 1 & 0 \end{bmatrix} \otimes A, \quad \mathbf{D} = (I - \mathbf{A})^{-\mathrm{T}} \mathbf{Q} (I - \mathbf{A})^{-1}, \tag{31}$$

where the positive-definite matrix Q is a covariance matrix, analogous to the "innovation noise" in the standard Kalman filter, $n \times n$ matrix A is a linear dynamics matrix[4].

We can then re-write the approximate posterior q as the product $q(\mathbf{z}|\mathbf{x}) \propto r_0(\mathbf{z}) r_1(\mathbf{z})$. By the standard product-of-normal-densities identity, $q(\mathbf{z}|\mathbf{x})$ also has a multivariate normal distribution $q = \mathcal{N}(\mu_\phi(\mathbf{x}), \Sigma_\phi(\mathbf{x}))$, where μ and Σ are given by eq. 19 (which we repeat here):

$$\Sigma_\phi(\mathbf{x}) = \left(\mathbf{D}^{-1} + \mathbf{C}_\phi^{-1}(\mathbf{x})\right)^{-1} \tag{32}$$
$$\mu_\phi(\mathbf{x}) = \left(\mathbf{D}^{-1} + \mathbf{C}_\phi^{-1}(\mathbf{x})\right)^{-1} \mathbf{C}_\phi^{-1}(\mathbf{x}) \mathbf{M}_\phi(\mathbf{x}). \tag{33}$$

Computation proceeds just as in Section 3.2.1, except that now the computation of eq. 19 takes the form,

$$\mu = (RR^{\mathrm{T}})^{-1} \mathbf{C}^{-1} M = R^{-\mathrm{T}}(R^{-1}(\mathbf{C}^{-1} M)) \tag{34}$$

which may be computed efficiently by exploiting the block bi-diagonal structure of R.

[4]For stable dynamics, we assume that the eigenvalues of A have magnitude less than one; in the examples we considered, we did not need to enforce this constraint.