Filtering Normalizing Flows

Hooshmand Shokri Razaghi
Department of Computer Science
Columbia University
New York, NY 10027
hooshmand@cs.columbia.edu

Liam Paninski
Department of Statistics
Columbia University
New York, NY 10027
liam@stat.columbia.edu

Abstract

Dynamical systems are the governing force behind many real world phenomena and temporally correlated data. Recently, a number of neural network architectures have been proposed to address inference for nonlinear dynamical systems. We introduce two different methods based on normalizing flows for posterior inference in latent non-linear dynamical systems. We also present gradient-based amortized posterior inference approaches using the auto-encoding variational Bayes framework that can be applied to a wide range of generative models with nonlinear dynamics. We call our method *Filtering Normalizing Flows* (FNF). FNF performs favorably against state-of-the-art inference methods in terms of accuracy of predictions and quality of uncovered codes and dynamics on synthetic data.

1 Introduction

Dynamical systems are the governing force behind many real-world phenomena and temporal data. For instance, in neuroscience, single single cell voltage recordings are modeled by a set of non-linear differential equations that are variants of the classical Hodgkin-Huxley neuron model while high-dimensional neural population activity are assumed to be noisy, redundant observations of latent and low dimensional signals, often modeled using state-space-models [12]. In recent years, a considerable number of generative models alongside inference algorithms have been proposed for neural population modelling. Among these, are temporal structure models [15, 17, 3], switching dynamical structure models [13, 10] and more recently neural network inspired generative models [1, 8, 2, 16, 4] that use auto-encoding variational Bayes (AEVB) [6] framework to amortize the inference of posterior distributions of latent representations.

In this work we consider a family of state-space models expressed by the following generative process, wherein, g, f are smooth differentiable functions, and $\Pi(\theta)$ is a noise distribution (e.g. Gaussian or Poisson respectively for continuous and discrete data) that is governed by parameters θ. The full joint distribution, denoted by $p(\mathbf{X}, \mathbf{Z})$, can be readily computed. This family subsumes well-know models such as *Linear Dynamical System* (LDS) and fLDS [1].

$$\begin{aligned} \mathbf{z}_1 &\sim \mathcal{N}(\mathbf{0}, Q_0) && \\ \mathbf{z}_t|\mathbf{z}_{t-1} &\sim \mathcal{N}\left(g(\mathbf{z}_{t-1}), Q\right) && t = 1, \dots, T \\ \mathbf{x}_t|\mathbf{z}_t &\sim \Pi(\theta = f(\mathbf{z_t})) && t = 1, \dots, T \end{aligned}$$

Normalizing flows[14] are expressive neural network based density estimators and have gained popularity recently. We propose two normalizing flows that are designed for the task of approximating posteriors of the aforementioned models. Also, we introduce an AEVB method to condition the

4th workshop on Bayesian Deep Learning (NeurIPS 2019), Vancouver, Canada.

parameters of the flow on observations such that any latent state of the approximate posterior $\mathbf{z}_t$ would depend on a sequence of observations $\mathbf{x}_{1..t+1}$ which is similar to particle filtering methods. Therefore, we name our method, *Filtering Normalizing Flows* (FNF).

2 Normalizing Flows

Normalizing flows are bijective transformations $f : \mathbb{R}^d \to \mathbb{R}^d$ that act on the space of continuous random variables in $\mathbb{R}^d$ to generate complex distributions using change of variables theorem given by equation 1 where $z \sim p(z)$ is a random variable with know distribution and and $f(z) = z' \sim q(z')$ is the new random variable induced by f.

$$\log q(z') = \log p(f^{-1}(z')) + \log \left| \frac{\partial f^{-1}}{\partial z'} \right| = \log p(z) - \log \left| \frac{\partial f}{\partial z} \right| \tag{1}$$

A category of transformations that contains the real NVP, MAF and IAF[7] have analytical inverses, apply the first hand side of the equation to model and optimize likelihoods directly. Whereas, for another category that includes planar flows the inverse is not analytically available. However, they can be used to construct variational approximation families $q_\phi(z)$ to estimate $\mathbb{E}_{q_\phi(z)}[L(z)]$ e.g. ELBO through Monte Carlo samples of q. In variational setting, application of these expressive densities is an attempt to decrease the gap between the variational objective and the log-likelihood and generally learn more desirable models and approximations of their posteriors.

Despite the existence of a variety normalizing flows, the task of applying them to structured models such as latent dynamical systems can be daunting. First, true posterior distributions of such models do not exhibit arbitrary correlation among the latent state space variables. Additionally, conditioning the latent states on observations is not straight forward since $\mathbf{z}_t|\mathbf{z}_{-t}$ is dependent on the entire sequence of observations $\mathbf{x}_{1:T}$. In section 3 we introduce two different transformations that condition latent variables on observation such that $\mathbf{z}_t|\mathbf{z}_{-t}$ is dependent on $\mathbf{x}_{1:t}$ that we call filtering. Furthermore, these models constrain the the flow such that the correlation in time induces smoothness in a way that the distribution does not factorize the same way as the prior, like the deep Kalman filter (DKF).

3 Filtering Normalizing Flows

Available normalizing flows, e.g. planar flows, are capable of expressing any arbitrary correlation in the space of $\mathbf{z} = \mathbf{z}_1\mathbf{z}_2 \cdots \mathbf{z}_T \in \mathbb{R}^{d \times T}$ when operating on the entire space. While we can apply any such flow on the entire state space, doing so is computationally expensive and conditioning on observations for amortized inference is not trivial. Therefore, we seek to constrain the expressiveness of our distribution globally while keeping it expressive enough locally so that the variational parameters admit a trivial choice for conditioning on observations.

3.1 Time Autoregressive

Consider the function $f : \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ with a set of parameters denoted by ϕ that transforms a consecutive a pair of latent states z_t, z_{t+1} and in doing so it correlates them with each other. We propose a flow that applies f sequentially to pairs $\mathbf{z}_{1:2}, \mathbf{z}_{2:3}, \ldots, \mathbf{z}_{T-1:T}$ with parameters $\phi_1, \phi_2 \ldots \phi_{T-1}$ respectively. In doing so, we hope to correlate all the latent variables in a smooth way with respect to time. Furthermore, as we see in section 4 this choice allows us to condition our latent trajectories on the observation in a straightforward way to achieve our filtering goal.

$$F_{\phi,t}(\mathbf{z}) = \begin{bmatrix} \mathbf{z}_{1:t-1} \\ f_\phi(\mathbf{z}_t, \mathbf{z}_{t+1}) \\ \mathbf{z}_{t+2:T} \end{bmatrix} \tag{2}$$

$$\left| \frac{\partial F_{\phi,t}}{\partial \mathbf{z}} \right| = \begin{vmatrix} I_{d(t-1)\times d(t-1)} & & \\ & \frac{\partial f}{\partial \mathbf{z_{t:t+1}}} & \\ & & I_{d(T-t-1)\times d(T-t-1)} \end{vmatrix} = \left| \frac{\partial f_\phi}{\partial \mathbf{z}_{t:t+1}} \right| \tag{3}$$

$$G(\mathbf{z}_{1:T}; \phi_1, \ldots, \phi_{T-1}) = F_{\phi_{T-1},T-1} \circ \cdots \circ F_{\phi_t,t} \circ \cdots \circ F_{\phi_1,1}(\mathbf{z}_{1:T}) \tag{4}$$

The composite flow G that is described by equation 4 subsumes the family of factorized models of form $q_1(\mathbf{z}) \prod_{t=2}^{T} q_t(\mathbf{z}_t|\mathbf{z}_{t-1})$ if $f(\mathbf{z}_t, \mathbf{z}_{t+1}) = \mathbf{z}_t, f(\mathbf{z}_{t+1})$ which contains the prior of the generative model. In this work we let $f(.)$ be multi-layer planar flows $f(\mathbf{z}_{t:t+1}) = \mathbf{z}_{t:t+1} + \mathbf{u}h(\mathbf{w}^T \mathbf{z}_{t:t+1} + b)$ which are appropriate choice for achieving expressiveness in low dimensionality of subspaces $\mathbf{z}_{t:t+1}$. For example, the choice of a single layer planar flow with time variant parameters $\{\mathbf{u}_t, \mathbf{w}_t, b_t\}$ gives us the recurrent solutions to the result of the transformation and the the probability that it induces on $\mathbf{z}_{1:T}$ in forms of equations 5 and 5.

$$\log q(\mathbf{z}') = \log p(\mathbf{z}) - \sum_{t=1}^{T-1} \log(1 + \mathbf{u}_t^\intercal \mathbf{w}_t h'(\mathbf{w}_t^\intercal \begin{bmatrix} \mathbf{z}_t'' \\ \mathbf{z}_{t+1} \end{bmatrix} + b_t)$$

Where $\mathbf{z}_t''$ is the intermediate transformation step from $\mathbf{z}_t$ to $\mathbf{z}_t'$ and is described by the equation below.

$$\mathbf{z}_t'' = \mathbf{z}_t + \mathbf{u}_{t-1} h'(\mathbf{w}_{t-1}^\intercal \begin{bmatrix} \mathbf{z}_{t-1}'' \\ \mathbf{z}_t \end{bmatrix} + b_{t-1}) \text{ for } t = 2, \ldots, T$$

3.2 Time Inverse Autoregressive

We also propose a second class of transformations described by equation 5 for achieving filtering property and constraining the resulting distribution for time-series data. This transformation is a combination of a fully connected layer and a skip layer where the weights of the fully connected layer form a lower/upper triangular matrix whose inverse is block-bi-diagonal. The rationale for this choice is threefold. First, the Jacobian is lower triangular, therefore, its determinant can be computed efficiently. Second, the inverse of $\mathbf{A}$ need not be computed for the forward transform since it can be solved efficiently and similarly it does not appear in log probability computation using determinant identity $|I + A^{-1}| = |A + I|/|A|$. Finally, this transformation imposes strong constraint on the correlation of the state space variables. This can be viewed as a nonlinear analogue of a Gaussian LDS's posterior samples being reparameterized by affine transformation of an inverse block-bi-diagonal matrix that is Cholesky factor of a block-tri-diagonal precision matrix of the true posterior.

$$\mathbf{z}'_{1:T} = f(\mathbf{z}; \mathbf{A}, \mathbf{b}, \mathbf{c}) = \mathbf{z}_{1:T} + \mathbf{c} \odot h(\mathbf{A}^{-1}\mathbf{z}_{1:T} + \mathbf{b}) \tag{5}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{D}_1 & & & \\ \mathbf{L}_1 & \mathbf{D}_2 & & \\ & \ddots & \ddots & \\ & & \mathbf{L}_{T-1} & \mathbf{D}`_T \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_T \end{bmatrix} \mathbf{c} = \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \vdots \\ \mathbf{c}_T \end{bmatrix} \tag{6}$$

$$\log \left| \frac{\partial f}{\partial \mathbf{z}} \right| = \sum_{t=1}^{T} \sum_{i=1}^{d} \log \left(1 + c_t(i) D_t(i,i) \psi_t(i)\right) - \sum_{t=1}^{T} \sum_{i=1}^{d} \log D_t(i,i) \tag{7}$$

In equation 7, $\boldsymbol{\psi} = h'(\mathbf{A}^{-1}\mathbf{z} + \mathbf{b})$, where $h(\cdot)$ is a smooth nonlinearity and $h'(\cdot)$ is its derivative. Unlike the previous method, we apply multiple layers of this transformation to construct expressive variational densities. We call this transformation an inverse filtering normalizing flow IFNF.

4 Auto-Encoding Variational Bayes

AEVB [6] is a variational framework for training deep generative models of type $p_\theta(\mathbf{x}, \mathbf{z})$ where there is a latent component $\mathbf{z}$. This method has been applied to many different settings including dynamical systems. AEVB optimizes the log likelihood $\mathbb{E}_{\mathbf{x} \sim Data}[p_\theta(\mathbf{x})]$ of the model, which is an

intractable marginal distribution, through optimizing a lower bound on it defined by equation 8 that requires introduction of $q_\phi(\mathbf{z}|\mathbf{x})$ a variational approximation to true posterior $p_\theta(\mathbf{z}|\mathbf{x})$.

$$\mathcal{L}(\theta, \phi; \mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}, \mathbf{z})] + H[q_\phi(\mathbf{z}|\mathbf{x})] \tag{8}$$

Furthermore, AEVB computes unbiased estimates of gradient of this objective with respect to the variational and model parameters θ, ϕ through reparameterization trick described by equation 9. This is achieved if samples from the variational distribution can be expressed as reparameterization of some noise samples. In other words $\mathbf{z} = f(\epsilon; \phi) \sim q_\phi(\mathbf{z})$ where $\epsilon \sim p_\epsilon(\epsilon)$ is noise.

$$\nabla_{\theta\phi} \mathbb{E}_{q_\phi(\mathbf{z})}[L_\theta(\mathbf{z})] = \mathbb{E}_{p_\epsilon(\epsilon)}[\nabla_{\theta\phi} L_\theta(f(\epsilon; \phi))] \tag{9}$$

ELBO is not a tight bound on the log likelihood because the family of the variational distributions do not contain the true posterior of the models in many cases. Normalizing flows are introduced to increase expressiveness of $q_\phi(\mathbf{z}|\mathbf{x})$ and therefore hoping to tighten the gap and train better models [14].

Our strategy in tackling the task of inference for dynamical systems is to rely on approximate evidence potentials coming from observations at each time steps denoted by $q_\phi^{(0)}(\mathbf{z}_t|\mathbf{x}_t)$. While these potentials can be described by normalizing flows themselves we simply use diagonal Gaussians $q_\phi^{(0)}(\mathbf{z}_t|\mathbf{x}_t) = \mathcal{N}(\mathbf{z}_t|\mu_\phi(\mathbf{x}_t), \sigma_\phi(\mathbf{x}_t))$. Our goal is to train our normalizing flows to mix local potentials with our prior and capture the true posterior better and ultimately learn a better model because of the expressiveness that our normalizing flows lend to our variational approximation.

While theoretically it is possible to mix local potentials without any further conditioning on the observations $\mathbf{x}_1 : T$ given the right invertible transformation, because of the limitations of the normalizing flows in general, we condition our normalizing flows on the observations as a proxy to gain more expressiveness. In the case of FNF time variant parameters $\phi_t(\mathbf{x}_t, \mathbf{x}_{t+1})$ are conditioned on pairs of consequitive observations through a neural network. In the case of IFNF, similarly we the parameters conditioned according to $\mathbf{D}_t(\mathbf{x}_t)$, $\mathbf{L}_t(\mathbf{x}_t, \mathbf{x}_{t+1})$, $\mathbf{b}_t(\mathbf{x}_t)$, $\mathbf{c}_t(\mathbf{x}_t)$. In our experiments we let the parameters of the flows to share networks with the local potential functions.

5 Experimental Results

In this section we compare the performance of our proposed method to that of fLDS and DKF which are two of the most widely used inference algorithms for latent dynamics models. We demonstrate through these experiments FNF/IFNF performs favourably against the state of the art in training deep dynamical systems while disentagling the dynamics from the nonlinear embeddings. For all the experiments we use DKF to refer to the inference network described in [8] as STLR that uses a bidirectional RNN for conditioning the factorized recognition model on the entire sequence of observations.

One quantitative way to asses the fit of state space models are forward extrapolations of the trained model given hold out observations. In order to do so we use the trained $q_\phi(\mathbf{z}|\mathbf{x})$ to sample L inferred trajectories $\mathbf{z}_{1:T}^{(l)}$ given held out observations. This is followed by evolving $\mathbf{z}_T^{(l)}$, the last states, according to the trained dynamics model $p_\theta(\mathbf{x}, \mathbf{z})$ K steps forward to obtain L trajectories in the observation space denoted by $\hat{\mathbf{x}}_{T:T+K}^{(l)}$. The metric that we use for the divergence of the forward extrapolation samples from true observations is described by 10. Lower divergence metrics show better extrapolations and indicate that the low-dimensional embedding of the emission model is disentangled from the dynamics of the transition model. Also, qualitatively we inspect these extrapolations and compare them with the model samples.

$$\text{MSE}_k = \frac{1}{L} \sum_{l=1}^{L} ||\mathbf{x}_{T+k} - \hat{\mathbf{x}}_{T+k}^{(l)}||_2 \tag{10}$$

We monitor and compare the rate of convergence of the evidence lower bound and its value to compare the fit of the models since higher ELBO potentially indicates higher log likelihoods for data.

5.1 Latent Lorenz System

The Lorenz system is a classical nonlinear differential equation in 3 independent variables.

$$\begin{aligned}\frac{d}{dt}z_1 =& \sigma(z_1 - z_2)\\ \frac{d}{dt}z_2 =& z_1(\rho - z_3) - z_2\\ \frac{d}{dt}z_3 =& z_1 z_2 - \beta z_3\end{aligned}$$

This is a well studied system with chaotic solutions that serves to clearly demonstrate FNF's advantage for inferring nonlinear dynamics. We generated Euler discretized numerical solutions for a stochastic Lorenz system (Gaussian additive noise of $\mathcal{N}(0, 0.1)$ at each step) from randomly generated initial solutions with parameters $\sigma = 10, \beta = \frac{8}{3}, \rho = 28$. For the purpose of this experiment we generate 100 trials where $T = 60, \mathbf{z_t} \in \mathbb{R}^3, \mathbf{x}_t = f(\mathbf{z}_t) + \epsilon \in \mathbb{R}^{10}$ where f is a single Layer MLP with softplus non-linearity with a skip layer connection to create smooth nonlinear expansions into observation space where $\epsilon \sim \mathcal{N}(0, 0.2)$. We train our models on %80 of the trails and report result on %20 validation set. We train only on 40 time steps and hold out 20 time steps at the end for comparing to extrapolations. We emphasize that stochastic noise exists in both the latent state-space evolution and the observation space. Figure 1 demonstrates how FNF is capable of learning complex nonlinear dynamics as well as non-linear embeddings by comparing kMSE of all methods and the quality of their extrapolations. As is demonstrated by figure 1, fLDS is not equipped with learning nonlinear dynamics and the extrapolations quickly diverge from the true values. DKF and FNF on the other hand are capable approximating and generalizing the non-linear dynamics and the embeddings. While this is the case, FNF outperforms DKF at this task since the extrapolations are qualitatively and quantitatively closer to the true values.

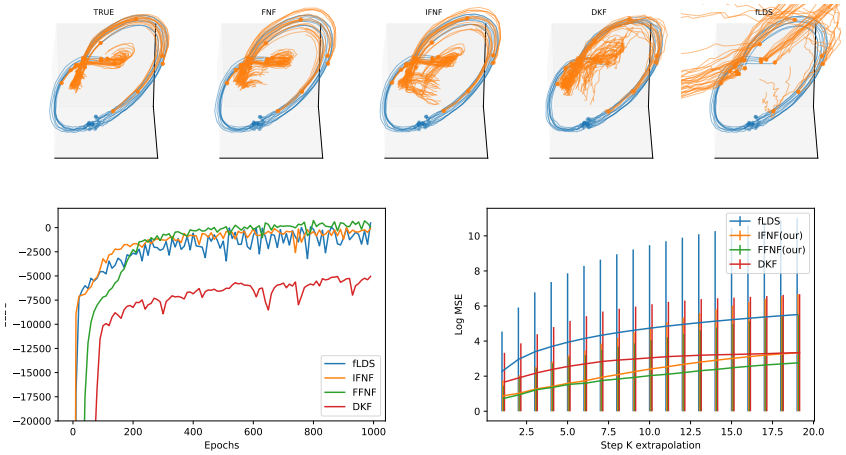

Figure 1: **Visualization of and observation trajectories for Lorenz system and performances on them.** (**top**) Respectively, true, FNF, IFNF, DKF, and fLDS reconstructions of the validation set (blue) and stochastic extrapolations into the future time steps (orange) for the true values orange is the hold out. (**bottom right**) kMSE of extrapolations using recovered latent state only. This demonstrate how well FNF disentangles hidden dynamics from embeddings. (**bottom left**) Rate of convergence of models are on par with each other, however FNF attains higher Bounds that could potentially translate into higher evidence values.

5.2 Rotating MNIST Images

In this task we showcase the ability of our method on synthetic task with and implicit underlying dynamics. Our synthetic dataset consists of 3500 randomly sampled MNIST digits that are binarized

and partitioned into 3000/500 for training/testing. Each image is rotated by $\alpha \sim \mathcal{N}(d \cdot \alpha, 2)$ degrees for 20 times where d is the value of the digit corresponding to the image and $\alpha = 18$ is the base rotation value. This way digit 0 rotates by noise and digit 9 rotates by 360 degrees and there digits in between proportional to their value. Beside the test set, we hold out 5 frames from each observation sequence to compare against forward extrapolations of the trained models. In this task we use a slightly different metric for the extrapolations which is the average likelihood of pixels under forecast samples given by equation 11.

$$\mathrm{L}_k = \frac{1}{N_{\text{pixels}}} \frac{1}{L} \sum_{N_{\text{pixels}}} \sum_{l=1}^{L} \text{Bernoulli}(\mathbf{x}_{T+k} | \hat{\mathbf{x}}_{T+k}^{(l)}) \tag{11}$$

This dataset is constructed to demonstrated how our model/inference is capable of recovering meaningful dynamics that translate into meaningful forward samples from the observations. We use two layer MLP with ReLU activation and 20 hidden units with skip layer connection for our transition model. The emission model is fixed across models with ta Bernoulli with rates conditioned on the latent states through MLP with ReLU activation and 50 hidden units for the emission model. The RNNs in DKF use 20 hidden units. Dimensionality of latent state space is set to 10 across all models. The recognition network of fLDS and FNF models use 2 layer MLP with ReLU activation. For FFNF IFNF we respectively use 5 and 3 layers of normalizing flow transformations that share parameters with the recognition network.

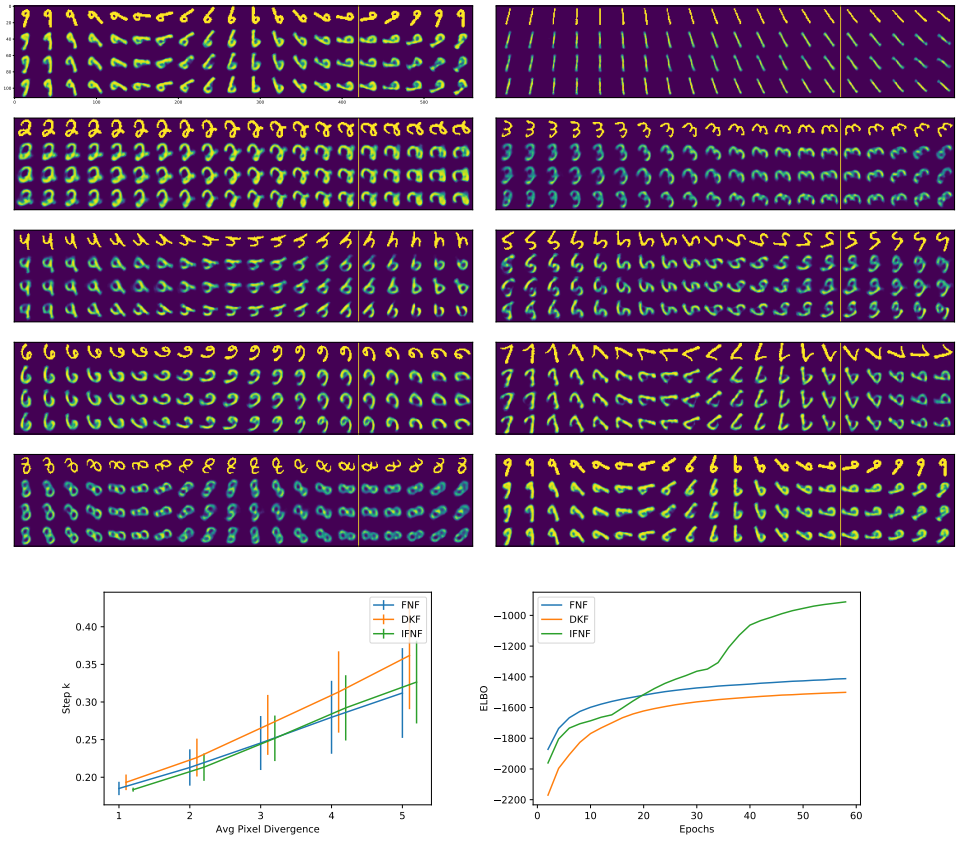

Figure 2: **Visualization of reconstructions and forward extrapolations for rotating MNIST images.** (**top**) 10 sequences of digits of test set being reconstructed and forward extrapolated (after solid yellow line). This demostrates how dynamics of rotation and latent representation of digits are separated since each digit has different rotaion value. Top row is true values and following that is respectively FNF, IFNF, and DKF. (**bottom left**) Inferred latent trajectory of digits for each model each color shows a different digit. 0 is blue and 1 is orange. (**bottom right**) ELBO of different models.

6 Related Work

Inference for state-space models that have underlying non-linearites is getting growing amounts of attention and some methods have been proposed for this task. Specially, since the introduction of AEVB[6] and its widespread success, many of these approaches have followed variational framework, and so does ours. The black box variational method that was introduced in [1] and extended to Poisson discrete observations in [2], known respectively as fLDS and PfLDS, consider a subset of the general class of latent dynamical systems with linear evolution in state space and nonlinear emissions. In other words, $\mathbf{z}_t|\mathbf{z}_{t-1} \sim \mathcal{N}(A\mathbf{z}_{t-1}, Q)$ governs latent state evolution. The variation approximation that fLDS uses is the result of mixing local independent Gaussian potentials $q_\phi(\mathbf{z_t}|\mathbf{x_t})$ with the prior of the model $p(\mathbf{z}_{1:T})$, the result of which is a Gaussian with a block-tri-diagonal precision matrix. Generalization of this method is offered by SVAE [5] which leverages message passing algorithm for graphical models with conjugate priors along with AEVB inference framework to mix local neural network potentials with a structured prior distribution. The conjugacy requirement of PGM inference limits the type of generative models allowed by this method to LDS, switching LDS, etc.

fLDS method is expanded by VIND [4] to use time variant linear dynamics to approximate non-linear dynamics. Learning the locally linear parameters are achieved through fixed point iterations. However, this method uses a Gaussian approximation that in non-linear dynamics regime can highly suffer from underestimation of variance that affect the fit of the model.

Deep Kalman Filter DKF and other structured inference networks [8] were proposed for inference for dynamical systems with non-linear transitions in the latent state space in the form of Gated Recurrent Units (GRU) with time variant noise. The variational posterior approximation is described by factorization $q_\phi(\mathbf{z}|\mathbf{x}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1}, \mathbf{x}_{1:t})$ which is conditioned on the observations either using MLP or RNN functions to filter/smooth. Among limitation of this model is using time invariant diagonal Gaussian noise and the challenges of training RNN and how slowly they propagate information for smoothing/filtering purposes.

In a slightly different approach of [9, 11] have taken a slightly different approach by using a optimizing a tighter bound on evidence of the generative model. They achieve this by conditioning a Sequential Monte Carlo (SMC) sampler on observations to simultaneously estimate the partition function that gives a better bound on log-likelihood and amortizing the posterior inference. However, due to re-sampling step in SMC gradients cannot be reparameterized and biased gradient estimates are used for optimization that poses variety of challenges. Also, this approach has the same problems as tighter lower bound approaches.

Finally, LFADS [16] proposes a sophisticated, bidirectional RNN architecture with neuroscience applications in mind, especially spike trains. The transition in this method functions are deterministic meaning that the underlying dynamics is not stochastic, rather it is superimposed with noise that does not affect the evolution.

References

[1] Evan Archer, Il Memming Park, Lars Buesing, John Cunningham, and Liam Paninski. Black box variational inference for state space models. *arXiv preprint arXiv:1511.07367*, 2016.

[2] Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical neural population models through nonlinear embeddings. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 163–171. Curran Associates, Inc., 2016.

[3] Robbe LT Goris, J Anthony Movshon, and Eero P Simoncelli. Partitioning neuronal variability. *Nature Neuroscience*, 2014.

[4] Daniel Hernandez, A. Moretti, Ziqiang Wei, S. Saxena, John Cunningham, and Liam Paninski. A novel variational family for hidden nonlinear markov models. *CoRR*, abs/1811.02459, 2018.

[5] Matthew Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. Composing graphical models with neural networks for structured representations and fast inference. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors,

Advances in Neural Information Processing Systems 29, pages 2946–2954. Curran Associates, Inc., 2016.

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

[7] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4743–4751. Curran Associates, Inc., 2016.

[8] Rahul G. Krishnan, Uri Shalit, and David Sontag. Deep kalman filters. *arXiv preprint arXiv:1511.05121*, 2016.

[9] Tuan Anh Le, Maximilian Igl, Tom Rainforth, Tom Jin, and Frank Wood. Auto-encoding sequential monte carlo. In *International Conference on Learning Representations*, 2018.

[10] Scott Linderman, Matthew Johnson, Andrew Miller, Ryan Adams, David Blei, and Liam Paninski. Bayesian Learning and Inference in Recurrent Switching Linear Dynamical Systems. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 914–922, Fort Lauderdale, FL, USA, 20–22 Apr 2017. PMLR.

[11] Christian A. Naesseth, Scott W. Linderman, Rajesh Ranganath, and David M. Blei. Variational sequential monte carlo. In *AISTATS*, 2018.

[12] L Paninski and JP Cunningham. Neural data science: accelerating the experiment-analysis-theory cycle in large-scale neuroscience. *Current Opinions in Neurobiology*, 50:232–241, 2018.

[13] Biljana Petreska, Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V Shenoy, and Maneesh Sahani. Dynamical segmentation of single trials from population neural data. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 756–764. Curran Associates, Inc., 2011.

[14] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1530–1538, Lille, France, 07–09 Jul 2015. PMLR.

[15] AC Smith and EN Brown. Estimating a state-space model from point process observations. *Neural Comput*, 15:965–991, 2003.

[16] David Sussillo, Rafal Józefowicz, L. F. Abbott, and Chethan Pandarinath. LFADS - latent factor analysis via dynamical systems. *CoRR*, abs/1608.06315, 2016.

[17] 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. *Journal of neurophysiology*, 102(1):614–35, Jul 2009.