Modelling Latent Dynamical Systems with Recognition-Parametrised Models

Samo Hromadka [1] **Maneesh Sahani** [1]

Abstract

We introduce a new approach to learning latent Markovian dynamical processes underlying observed time series data: the recognition-parametrised latent dynamical system (RP-LDS). The RP-LDS resolves issues in two broad classes of state-of-the-art latent time series models, while maintaining expressivity through a complex neural network-based link between observations and latents. As opposed to *generative* or auto-encoding approaches, the RP-LDS does not learn an explicit model reconstructing observations from latents, thus allowing it to avoid parameter bias and focus model capacity on recognition. As opposed to *contrastive* approaches, the RP-LDS utilises efficient message-passing to propagate posterior uncertainty and achieve maximum-likelihood learning. The RP-LDS matches the performance of state-of-the-art methods on both linear and nonlinear toy problems. We apply the RP-LDS to video of a swinging pendulum with background distractors and show that it is able to recover the underlying latent system despite not being in model class.

1. Introduction

Unsupervised representation learning from time series data is crucial in many applications, such as reinforcement learning, robotics, navigation, or signal processing. Under strict assumptions, learning and inference can be done exactly, most famously via the EM algorithm with Kalman smoothing (Kalman, 1960; Neal & Hinton, 1998). In more general settings, approximations must be made.

One approach to time-series representation learning is *generative*, in which generative and recognition networks are trained jointly. Recent years have seen an increase in the use of variational autoencoders (VAEs; Kingma & Welling, 2014) adjusted to handle time series data. One of the first successful VAE-based methods was the Deep Kalman Filter (DKF; Krishnan et al., 2015). The DKF parametrises the variational posterior as a recurrent neural network (RNN) that maps data to a distribution over latents. Another approach is the Structured VAE (SVAE; Johnson et al., 2016), which learns a recognition network that returns conjugate potentials that allow for exact message-passing inference. The SVAE has recently been optimised for efficient training on GPUs and shown to be successful across a range of tasks (Zhao & Linderman, 2023).

Generative approaches learn explicit generative networks, even though many downstream applications only require recognition. This may lead to unnecessary error in the model, as generative models that are sufficiently complex to model real-world data do not admit exact inference and must thus resort to approximations. Approximate methods can yield biased parameter estimates (Turner & Sahani, 2011). Flexible generative models can further degrade recognition networks by compensating for approximations in the variational posterior (Cremer et al., 2018).

Another approach is *contrastive*. Contrastive models bypass the need to train an explicit generative model and instead learn a recognition network by contrasting "positive" data points from "negative" ones, typically with regularisation to avoid collapse. A commonly used contrastive approach for time series data is InfoNCE (van den Oord et al., 2018).

However, by not performing maximum-likelihood in a concrete probabilistic model, contrastive methods are not able to use variational tools from the probabilistic graphical model literature and cannot provide principled posterior beliefs over latent variables; a feature that is crucial to optimal Bayesian decision making. Recent extensions of contrastive methods do provide a form of posterior uncertainty, but not in a maximum-likelihood framework (Kirchhof et al., 2023). Despite success across many domains, contrastive methods have been found to not be sufficiently expressive in some applications, such as model-based reinforcement learning (Hafner et al., 2020).

Recent work introduced a class of semi-parametric models called recognition-parametrised models (RPMs; Walker et al., 2023), which do not learn an explicit generative model

[1]Gatsby Computational Neuroscience Unit, University College London, London, UK. Correspondence to: Samo Hromadka <s.hromadka@ucl.ac.uk>.

Accepted by the Structured Probabilistic Inference & Generative Modeling workshop of ICML 2024, Vienna, Austria.

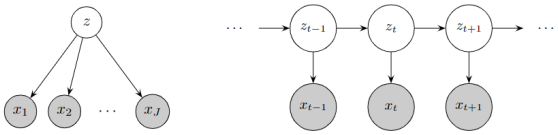


Figure 1: **Left:** RPM graphical model. **Right:** LDS graphical model.

but nonetheless offer consistent estimation for an implicit likelihood (Author, 2021). RPMs were shown to be successful in a range of settings. In this work, we aim to address shortcomings of both generative and contrastive methods for time series data by applying RPMs to latent dynamical systems (summarised in Table 1).

Table 1: Properties of different methods.

Method	Generation not required	Maximum likelihood	Posterior uncertainty
RPM	✓	✓	✓
VAE	✗	✓	✓
NCE	✓	✗	✗

2. Background

We first introduce RPMs (Walker et al., 2023) in their most general form, then detail the framework of latent dynamical systems. We assume continuous latents throughout.

2.1. The Recognition-Parametrised Model

The RPM is a model for unsupervised representation learning that exploits conditional independence relationships between latent and observed variables. Given a graphical model as in Figure 1 (left), the joint distribution between latent and observed variables is

$$p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{z}) \prod_{j=1}^{J} p(x_j|\boldsymbol{z}). \tag{1}$$

The index j can be thought of as corresponding to multiple modalities, e.g. image, sound, and text, generated by a common latent state. We use bold font for the latent $\boldsymbol{z}$ to emphasise that the latents can have arbitrary graphical structure. The RPM avoids the explicit generative model in Equation (1) by applying Bayes' rule and approximating the resulting terms:

$$p_{\boldsymbol{X}}(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{z}) \prod_{j=1}^{J} \frac{f_{\phi_j}(\boldsymbol{z}|x_j) p_0(x_j)}{F_{\phi_j}(\boldsymbol{z})}, \tag{2}$$

where:

- $\boldsymbol{X} = \{x_j^n\}_{j=1,n=1}^{J,N}$ is a dataset of N data points for each modality j;
- p_0 is the empirical distribution on $\boldsymbol{X}$: $p_0(x_j) = \frac{1}{N}\sum_{n=1}^{N} \delta(x_j - x_j^n)$;
- $f_{\phi_j}(\boldsymbol{z}|x_j)$ are parametrised *recognition factors*;
- $F_{\phi_j}(\boldsymbol{z})$ ensures that the RPM joint in Equation (2) is normalised:

$$F_{\phi_j}(\boldsymbol{z}) = \int f_{\phi_j}(\boldsymbol{z}|x_j) p_0(x_j)\, \mathrm{d}x_j = \frac{1}{N}\sum_{n=1}^{N} f_{\phi_j}(\boldsymbol{z}|x_j^n).$$

The RPM joint is subscripted by the dataset because the joint is itself a function of the dataset through its dependence on p_0. We parametrise the prior with parameters η, so that the total parameter set is $\theta := (\eta, \{\phi_j\}_{j=1}^{J})$ and the parametrised RPM joint is denoted by $p_{\theta,\boldsymbol{X}}(\boldsymbol{x}, \boldsymbol{z})$. The RPM is fit via the EM algorithm, corresponding to coordinate ascent on the free energy of the RPM joint (Neal & Hinton, 1998). In the case of continuous latent variables, additional approximations must be made because $F_{\phi_j}(\boldsymbol{z})$ is a mixture distribution, rendering the term $\langle \log F_{\phi_j}(\boldsymbol{z}) \rangle$ in the free energy intractable. We detail these further in Section 3.

2.2. Latent Dynamical System Models

Latent dynamical system (LDS) models have graphical structure as in Figure 1 (right). The joint distribution and free energy are

$$p_\theta(\boldsymbol{x}, \boldsymbol{z}) = p_\theta(z_1) \prod_{t=2}^{T} p_\theta(z_t|z_{t-1}) \prod_{t=1}^{T} p_\theta(x_t|z_t), \tag{3}$$

$$\mathcal{F}(q, \theta) = \langle \log p_\theta(\boldsymbol{x}, \boldsymbol{z}) \rangle_{q(\boldsymbol{z})} + \mathcal{H}(q), \tag{4}$$

where $\mathcal{H}$ denotes the entropy of a distribution. Learning is done via the EM algorithm. The E-step, which updates q given fixed θ, can be computed in closed form in specific cases. The most famous case consists of linear-Gaussian transition and emission distributions, which is solved by Kalman smoothing (Kalman, 1960).

3. Method

We assume a standard LDS graphical model as in Figure 1 (right) with a single emission x_t at each time, although our methods easily extend to multiple emissions at each time conditionally independent of z_t. The RPM joint and free energy are

$$p_{\theta,\boldsymbol{X}}(\boldsymbol{x}, \boldsymbol{z}) = p_\eta(z_1) \prod_{t=2}^{T} p_\eta(z_t|z_{t-1}) \prod_{t=1}^{T} \frac{f_\phi(z_t|x_t) p_0(x_t)}{F_\phi(z_t)}, \tag{5}$$

$$\mathcal{F}(\{q^n\},\theta) \overset{+c}{=} \sum_{n=1}^{N}\Bigg[\mathcal{H}(q^n) + \langle \log p_\eta(z_1)\rangle_{q^n(z_1)} + \sum_{t=2}^{T}\langle \log p_\eta(z_t|z_{t-1})\rangle_{q^n(z_t,z_{t-1})} + \sum_{t=1}^{T}\left\langle \log \frac{f_\phi(z_t|x_t^n)}{F_\phi(z_t)}\right\rangle_{q^n(z_t)}\Bigg]. \quad (6)$$

Although we parametrise the prior and recognition factors as jointly Gaussian in $\boldsymbol{z}$, the inclusion of arbitrarily nonlinear links between observations and latents makes if possible to model complex, real-world time series:

$$\begin{aligned} p_\eta(z_1) &= \mathcal{N}(z_1|m_1, Q_1),\\ p_\eta(z_t|z_{t-1}) &= \mathcal{N}(z_t|A_t z_{t-1} + b_t, Q_t),\\ f_\phi(z_t|x_t) &= \mathcal{N}(z_t|\mu_\phi(x_t), \Sigma_\phi(x_t)). \end{aligned}$$

We adapt approximations from Walker et al. (2023) to perform EM on the RPM free energy.

3.1. M-step

The terms $F_\phi(z_t)$ are mixtures of Gaussians, so $\langle \log F_\phi(z_t)\rangle_{q^n(z_t)}$ are intractable. To resolve this we use the *interior variational bound* (Walker et al., 2023): by Jensen's inequality,

$$-\langle \log F_\phi(z_t)\rangle_{q^n(z_t)} \geq -\left\langle \log \frac{q^n(z_t)}{\tilde{f}^n(z_t)}\right\rangle_{q^n(z_t)} - \log \Gamma_{\phi,t}^n,$$

where the $\tilde{f}^n$ are arbitrary *auxiliary factors* and $\Gamma_{\phi,t}^n = \int F_\phi(z_t)\tilde{f}^n(z_t)\,\mathrm{d}z_t$. From Jensen's inequality, the optimal value of $\tilde{f}^n$ is $\tilde{f}^n(z_t) \propto q^n(z_t)/F_\phi(z_t)$. We use the ansatz $F_\phi(z_t) \to p_\eta(z_t)$ (Walker et al., 2023) to set $\tilde{f}^n(z_t) \propto q^n(z_t)/p_\eta(z_t)$. Finally, we decompose the recognition factors into a time-invariant component $\bar{f}_\phi$ and a potentially time-varying component given by the prior: $f_\phi(z_t|x_t) \propto p_\eta(z_t)\bar{f}_\phi(z_t|x_t)$. This parametrisation alleviates the need to fit T separate recognition networks when the latent dynamics are nonstationary. It also ensures that all terms $\Gamma_{\phi,t}^n$ are finite, as shown in Appendix A.

The resulting lower bound on the free energy can be rewritten as

$$\tilde{\mathcal{F}}(\{q^n\},\theta) = \sum_{n=1}^{N}\Bigg[\sum_{t=1}^{T}\left\{\log \hat{\Gamma}_{\theta,t}^n - \mathsf{KL}\left(q^n(z_t)\,\middle\|\,\hat{f}_\phi^n(z_t)\right)\right\} - \mathsf{KL}\left(q^n(\boldsymbol{z})\,||\,p_\eta(\boldsymbol{z})\right)\Bigg], \quad (7)$$

where $\hat{f}_\phi^n(z_t|x_t^n) \propto \bar{f}_\phi(z_t|x_t^n)q^n(z_t)$ is a normalised distribution and the $\hat{\Gamma}_{\theta,t}^n$ are terms depending on $\bar{f}_\phi$, p_η, and q^n. A full derivation is given in Appendix A. The M-step proceeds by gradient ascent of $\tilde{\mathcal{F}}$ on θ.

3.2. E-step

The E-step is performed on an approximation of $\mathcal{F}$, rather than directly on $\tilde{\mathcal{F}}$. Noting the similarity between Equations (3) and (5), the E-step on $\mathcal{F}$ can be computed with a Kalman smoother, but with the usual emissions $p(x_t|z_t)$ replaced by $f_\phi(z_t|x_t)/F_\phi(z_t)$. As $F_\phi(z_t)$ is a mixture, computing the smoothing messages is intractable. We approximate $F_\phi(z_t) \approx p_\eta(z_t)$ and use the parametrisation of f_ϕ from the M-step to get $f_\phi(z_t|x_t)/F_\phi(z_t) \approx \bar{f}_\phi(z_t|x_t)$. The E-step can then be computed with a Kalman smoother with emissions proportional to $\bar{f}_\phi(z_t|x_t)$. Crucially, because $\bar{f}_\phi(z_t|x_t)$ is linear in z_t, Kalman smoothing is exact, despite a nonlinear neural network-based link between z_t and x_t.

4. Experiments

We demonstrate the effectiveness of the RP-LDS on simulated problems of increasing difficulty. We begin with two toy problems with linear ground-truth dynamical systems and linear and nonlinear emission functions, respectively. We then show that the RP-LDS is able to recover latent variables describing pendulum motion, a fundamentally nonlinear system, from image data. To highlight the advantage of not having an explicit generative model, we also apply the RP-LDS to pendulum image data with background distractors.

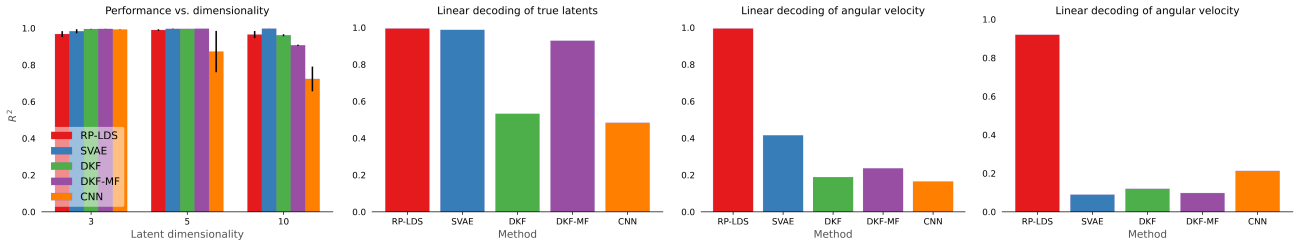
It is difficult to compare the RP-LDS to other methods via free energy, as the RP-LDS free energy is a function of the data and will in general vary when the data changes. We instead measure performance by the R^2 value of linear regression between each model's inferred latent variables (posterior means) and the ground-truth latent variables.

We compare the RP-LDS to the SVAE (Johnson et al., 2016; Zhao & Linderman, 2023) and the DKF (Krishnan et al., 2015) with three different parametrisations of the variational posterior family: two based on bidirectional RNNs and one based on a CNN with convolutions over the time dimension. We refer to these parametrisations as DKF, DKF-MF, and CNN, respectively. Full parametrisation and experimental details can be found in Appendix B. Code for all models and experiments is heavily inspired by that of Zhao & Linderman (2023).

4.1. Linear Dynamical System with Linear Emissions

To test the performance of the RP-LDS in model class we simulate latents and observations from random linear-Gaussian systems:

$$z_1 \sim \mathcal{N}(\mu_1, S_1), \quad (8)$$
$$z_t|z_{t-1} \sim \mathcal{N}(Bz_{t-1}, S), \quad (9)$$
$$x_t|z_t \sim \mathcal{N}(Cz_t + d, R). \quad (10)$$



(a) Linear (b) Swiss roll (c) Pendulum (d) CIFAR-pendulum

Figure 2: **(a):** the RP-LDS maintains competitive performance on the linear system with linear emissions. Results are averaged over three seeds and error bars indicate standard error. **(b):** the RP-LDS matches the performance of SVAE on the linear system with Swiss roll emission. **(c):** the RP-LDS is the only method able to decode pendulum angular velocity when data masking is not used. **(d):** the RP-LDS is the only method able to decode CIFAR-pendulum angular velocity, even when other methods are trained with data masking.

As in Zhao & Linderman (2023), we test performance on three pairs of latent and observation dimensionalities: $\{(3, 5), (5, 10), (10, 20)\}$. As shown in Figure 2(a), the RP-LDS shows comparable performance to state-of-the-art methods across the three dimensionalities.

4.2. Linear Dynamical System with Nonlinear Emissions

In the second experiment, we take a latent system as in Equations (8) and (9) with two-dimensional latents, which are mapped to three-dimensional observations via the "Swiss roll" function (Tenenbaum et al., 2000) plus Gaussian noise. A full description of the nonlinearity can be found in Appendix B.2. Results are shown in Figure 2(b); RP-LDS and SVAE both achieve R^2 very near to 1.

4.3. Pendulum Video Task

Next, we apply the RP-LDS to video data of a simulated pendulum (Becker et al., 2019). The pendulum's dynamics are governed by angle and angular velocity, which constitute a two-dimensional nonlinear dynamical system. This task thus tests the ability to learn from images and to learn from data with nonlinear latent dynamics. Whereas angle can be decoded from individual frames, decoding angular velocity requires both dynamics and recognition models to be accurate.

To showcase that learning an explicit generative model can sometimes be harmful, we also construct a dataset of pendulum videos where each frame's background is given by a randomly chosen image from the CIFAR-10 dataset (Krizhevsky & Hinton, 2009). A generative model would aim to reconstruct the entire images, which is incredibly difficult, especially as there are no structured temporal dynamics between the images.

We run all methods with three-dimensional latents. Figure 2(c-d) shows R^2 values of linearly decoding angular velocity in the pendulum and CIFAR-pendulum datasets, respectively. Results for decoding angle are given in Appendix B.3. In both datasets, the RP-LDS is the only method that can reliably linearly decode angular velocity. Zhao & Linderman (2023) showed that the SVAE with three-dimensional latents can decode angular velocity well, but this crucially depends on masking a fraction of the data to force the SVAE to learn a good dynamics model. When data is not masked, as in Figure 2(c), the generative model of the SVAE is so expressive that there is little incentive to learn a good dynamics model. In contrast, the lack of generative model in the RP-LDS allows for a good dynamics model to be learned. For the CIFAR-pendulum dataset we train all methods other than RP-LDS with data masking as in Zhao & Linderman (2023), but they are still unable to linearly decode angular velocity due to the complex image backgrounds.

5. Conclusion

We have introduced the *recognition-parametrised latent dynamical system*, a probabilistic method for learning and inference in latent time series. The RP-LDS addresses shortcomings of existing generative and contrastive methods: it has no explicit generative model, it is asymptotically maximum-likelihood, and it provides principled posterior uncertainty via efficient message-passing. We show that the RP-LDS approximately matches the performance of existing methods on simple toy problems with linear ground-truth latent dynamics. On a problem with high-dimensional observations and nonlinear ground-truth latent dynamics, the RP-LDS is able to simultaneously learn good dynamics and recognition models without the need to mask data.

The RP-LDS is a promising method to learn latent dynamics from time series, particularly for downstream tasks that do not require data generation. Our results indicate that the RP-LDS could be a very competitive method in problems such as model-based reinforcement learning.

References

Author, N. N. Suppressed for anonymity, 2021.

Becker, P., Pandya, H., Gebhardt, G., Zhao, C., Taylor, J., and Neumann, G. Recurrent Kalman networks: Factorized inference in high-dimensional deep feature spaces. *arXiv*, abs/1905.07357, 2019.

Cremer, C., Li, X., and Duvenaud, D. Inference suboptimality in variational autoencoders. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1078–1086. PMLR, 10–15 Jul 2018.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. *arXiv*, abs/1912.01603, 2020.

Johnson, M. J., Duvenaud, D. K., Wiltschko, A., Adams, R. P., and Datta, S. R. Composing graphical models with neural networks for structured representations and fast inference. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

Kalman, R. E. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82 (1):35–45, 03 1960. ISSN 0021-9223. doi: 10.1115/1.3662552.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Kirchhof, M., Kasneci, E., and Oh, S. J. Probabilistic contrastive learning recovers the correct aleatoric uncertainty of ambiguous inputs. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 17085–17104. PMLR, 23–29 Jul 2023.

Krishnan, R. G., Shalit, U., and Sontag, D. Deep Kalman filters. *arXiv*, abs/1511.05121, 2015.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Ontario, 2009.

Neal, R. M. and Hinton, G. E. *A View of the EM Algorithm that Justifies Incremental, Sparse, and other Variants*, pp. 355–368. Springer Netherlands, Dordrecht, 1998. ISBN 978-94-011-5014-9. doi: 10.1007/978-94-011-5014-9_12.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. doi: 10.1126/science.290.5500.2319.

Turner, R. E. and Sahani, M. *Two problems with variational expectation maximisation for time series models*, pp. 104–124. Cambridge University Press, 2011.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018.

Walker, W. I., Soulat, H., Yu, C., and Sahani, M. Unsupervised representation learning with recognition-parametrised probabilistic models. *arXiv*, abs/2209.05661, 2023.

Zhao, Y. and Linderman, S. Revisiting structured variational autoencoders. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 42046–42057. PMLR, 23–29 Jul 2023.

A. Loss Derivation

We derive the lower bound in Equation (7) for general exponential family distributions.

Recall that the RPM free energy is

$$\mathcal{F}(\{q^n\},\theta) \stackrel{+c}{=} \sum_{n=1}^{N}\Bigg[\langle \log p_\eta(z_1)\rangle_{q^n(z_1)} + \sum_{t=2}^{T}\langle \log p_\eta(z_t|z_{t-1})\rangle_{q^n(z_t,z_{t-1})} + \sum_{t=1}^{T}\left(\langle \log f_\phi(z_t|x_t^n)\rangle_{q^n(z_t)} - \langle \log F_\phi(z_t)\rangle_{q^n(z_t)}\right) + \mathcal{H}(q^n)\Bigg] \tag{A.1}$$

and the interior variational bound is

$$-\langle \log F_\phi(z_t)\rangle_{q^n(z_t)} \geq -\left\langle \log \frac{q^n(z_t)}{\tilde{f}^n(z_t)}\right\rangle_{q^n(z_t)} - \log \Gamma_{\phi,t}^n. \tag{A.2}$$

Omit parameter subscripts for clarity. Assume that $f(z_t|x_t^n)$, $q^n(z_t)$, and $p(z_t)$ all belong to the same exponential family with natural parameters $\eta(x_t^n)$, η_{qt}^n, and η_{0t}, respectively:

$$\begin{aligned} f_\phi(z_t|x_t^n) &= h(z_t)e^{\eta(x_t^n)^\top t(z_t) - \Phi(\eta(x_t^n))},\\ q^n(z_t) &= h(z_t)e^{(\eta_{qt}^n)^\top t(z_t) - \Phi(\eta_{qt}^n)},\\ p(z_t) &= h(z_t)e^{\eta_{0t}^\top t(z_t) - \Phi(\eta_{0t})}. \end{aligned}$$

Let $\tilde{f}^n(z_t)$ be a general factor with the same shape and its own natural parameter $\tilde{\eta}_t^n$:

$$\tilde{f}^n(z_t) = e^{(\tilde{\eta}_t^n)^\top t(z_t)}.$$

Then the terms Γ_t^n from Equation (A.2) become

$$\begin{aligned} \Gamma_t^n &= \frac{1}{N}\sum_{n'=1}^{N}\int h(z_t)e^{\left(\eta\left(x_t^{n'}\right)+\tilde{\eta}_t^n\right)^\top t(z_t) - \Phi\left(\eta\left(x_t^{n'}\right)\right)}\,\mathrm{d}z_t\\ &= \frac{1}{N}\sum_{n'=1}^{N} e^{\Phi\left(\eta\left(x_t^{n'}\right)+\tilde{\eta}_t^n\right) - \Phi\left(\eta\left(x_t^{n'}\right)\right)} \end{aligned}$$

Next, using Equation (A.2),

$$\left\langle \log \frac{f(z_t|x_t^n)}{F(z_t)}\right\rangle_{q^n(z_t)} \geq \left\langle \log \frac{f(z_t|x_t^n)\tilde{f}^n(z_t)}{q^n(z_t)}\right\rangle_{q^n(z_t)} - \log \Gamma_t^n.$$

Defining $\hat{f}^n(z_t)$ to belong to the same exponential family with natural parameter $\eta(x_t^n)+\tilde{\eta}_t^n$ and adding and subtracting,

$$\begin{aligned} \left\langle \log \frac{f(z_t|x_t^n)}{F(z_t)}\right\rangle_{q^n(z_t)} &\geq \left\langle \log \frac{f(z_t|x_t^n)\tilde{f}^n(z_t)\hat{f}^n(z_t)}{q^n(z_t)\hat{f}^n(z_t)}\right\rangle_{q^n(z_t)} - \log \Gamma_t^n\\ &= -\mathsf{KL}\left(q^n(z_t)\,\middle\|\,\hat{f}^n(z_t)\right) + \left\langle \log \frac{f(z_t|x_t^n)\tilde{f}^n(z_t)}{\hat{f}^n(z_t)}\right\rangle_{q^n(z_t)} - \log \Gamma_t^n\\ &= -\mathsf{KL}\left(q^n(z_t)\,\middle\|\,\hat{f}^n(z_t)\right) - \Phi(\eta(x_t^n)) + \Phi(\eta(x_t^n)+\tilde{\eta}_t^n) - \log \Gamma_t^n\\ &= \log \hat{\Gamma}_t^n - \mathsf{KL}\left(q^n(z_t)\,\middle\|\,\hat{f}^n(z_t)\right) + \log N, \end{aligned}$$

where

$$\hat{\Gamma}_t^n = \frac{e^{\Phi(\eta(x_t^n)+\tilde{\eta}_t^n) - \Phi(\eta(x_t^n))}}{\sum_{n'=1}^{N} e^{\Phi\left(\eta\left(x_t^{n'}\right)+\tilde{\eta}_t^n\right) - \Phi\left(\eta\left(x_t^{n'}\right)\right)}}.$$

The parametrisations detailed in Section 3.1 are equivalent to the following parametrisations in natural parameter space:

$$\eta(x_t^n) = \eta_{0t} + \bar{\eta}(x_t^n),$$
$$\tilde{\eta}_t^n = \eta_{qt}^n - \eta_{0t},$$

where $\bar{\eta}(x_t^n)$ are the natural parameters corresponding to $\bar{f}(z_t|x_t^n)$. This ensures that $\eta(x_t^n)$ and $\eta\left(x_t^{n'}\right) + \tilde{\eta}_t^n$ are valid natural parameters for all n, n' and t.

Applying this bound for all n and t, we obtain a lower bound to the free energy from Equation (A.1):

$$\begin{aligned}
\mathcal{F}(\{q^n\}, \theta) &\geq \tilde{\mathcal{F}}(\{q^n\}, \theta) \\
&= \sum_{n=1}^{N} \Bigg[\sum_{t=1}^{T} \left\{ \langle \log p(z_t|z_{t-1}) \rangle_{q^n(z_t, z_{t-1})} + \log \hat{\Gamma}_t^n - \mathsf{KL}\left(q^n(z_t) \,\middle\|\, \hat{f}^n(z_t)\right) \right\} \\
&\qquad + \langle \log p(z_1) \rangle_{q^n(z_1)} + \mathcal{H}(q^n) \Bigg] \\
&= \sum_{n=1}^{N} \left[\sum_{t=1}^{T} \left\{ \log \hat{\Gamma}_t^n - \mathsf{KL}\left(q^n(z_t) \,\middle\|\, \hat{f}^n(z_t)\right) \right\} - \mathsf{KL}\left(q^n(\boldsymbol{z}) \,||\, p(\boldsymbol{z})\right) \right].
\end{aligned}$$

B. Experimental Details

In all experiments we have a separate learning rate over prior parameters and the remaining model parameters, i.e. for recognition and generative networks, as appropriate. We refer to these learning rates as prior and base learning rates, respectively. Results are computed with a grid hyperparameter search over both learning rates taking values in $\{10^{-3}, 10^{-2}\}$.

We use a periodic cosine schedule for the prior learning rate and a linear warmup followed by constant schedule for the base learning rate. All experiments are run with a batch size of 10 and a total of $N = 100$ sequences. Linear and Swiss roll experiments are run with $T = 200$ and the pendulum experiments are run with $T = 100$.

The three DKF posterior families, which are the same as in Zhao & Linderman (2023), are described below.

- **DKF:**
$$q_\phi(z_{1:T}) = \mathcal{N}(z_1|m_{\phi,1}, S_{\phi,1}) \prod_{t=2}^{T} \mathcal{N}(z_t|A_{\phi_t} z_{t-1} + m_{\phi,t}, S_{\phi,t}),$$
where $\{m_{\phi,t}, A_{\phi,t}, S_{\phi,t}\}_{t=1}^T$ are the outputs of a bidirectional RNN with weights ϕ applied to the data $x_{1:T}$.
- **DKF-MF:**
$$q_\phi(z_{1:T}) = \prod_{t=1}^{T} \mathcal{N}(z_t|m_{\phi,t}, S_{\phi,t}),$$
where $\{m_{\phi,t}, S_{\phi,t}\}_{t=1}^T$ are the outputs of a bidirectional RNN with weights ϕ applied to the data $x_{1:T}$.
- **CNN:** the same as DKF, but with the parameters $\{m_{\phi,t}, A_{\phi,t}, S_{\phi,t}\}_{t=1}^T$ given by the output of a CNN with temporal convolutions applied to the data $x_{1:T}$.

B.1. Linear

Inspired by Zhao & Linderman (2023), we define B to be a rotation matrix around a random axis, choose S_1, S, and R to be equal to $0.1I$, and sample all components of C and d i.i.d. from $\mathcal{N}(0, 1)$.

B.2. Swiss Roll

In the Swiss roll experiment all recognition/generation networks are taken to be MLPs with ReLU nonlinearities. Results are computed with a further grid search over the networks having 1 or 2 layers. In either case, the number of hidden neurons in each layer is 5.

The Swiss roll nonlinearity $S : \mathbb{R}^2 \to \mathbb{R}^3$ is given by

$$S(x, y) = \begin{pmatrix} u(x)\sin(4\pi u(x)) \\ u(x)\cos(4\pi u(x)) \\ u(y) \end{pmatrix},$$

where $u : \mathbb{R} \to \mathbb{R}$ is given by

$$u(x) = \frac{\tanh(x/10) + 1}{2}.$$

Figure 3 illustrates the Swiss roll nonlinearity without and with noise.

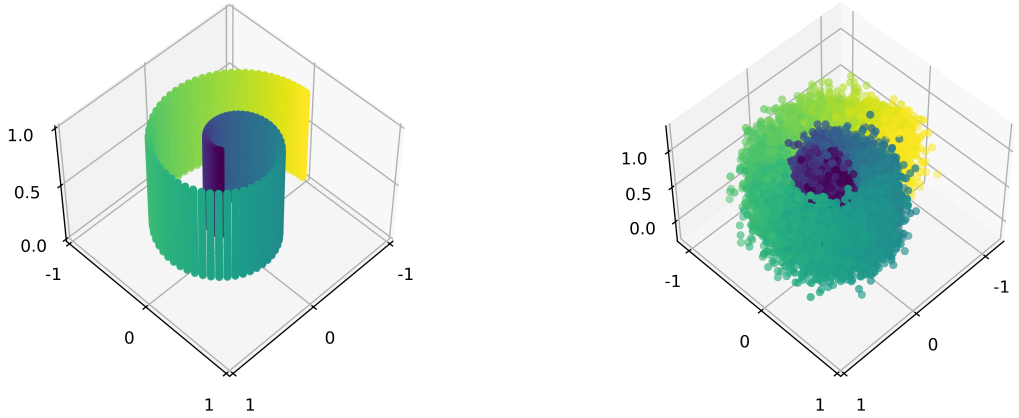
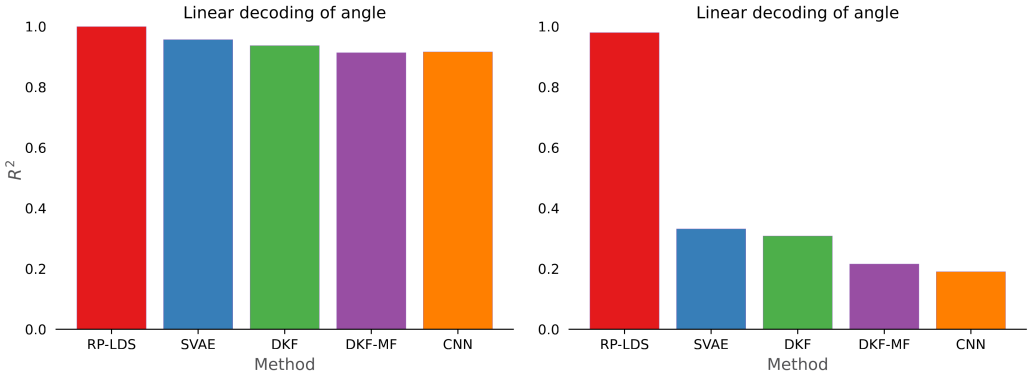


Figure 3: **Left:** the Swiss roll function image. **Right:** the Swiss function image plus Gaussian noise.

B.3. Pendulum

Figure 4 shows the linear R^2 scores for decoding pendulum and CIFAR-pendulum angle across all methods. All methods recover the angle well on the pendulum dataset, even without data masking (even though they fail to recover angular *velocity* without masking; Figure 2). Only RP-LDS is able to identify the angle time series on the CIFAR-pendulum dataset, even when other methods are trained with data masking.



(a) Pendulum (b) CIFAR-pendulum

Figure 4: **(a):** RP-LDS maintains competitive performance on linearly decoding pendulum angle. Other methods are trained without data masking. **(b):** RP-LDS significantly outperforms other methods in linearly decoding pendulum angle when CIFAR-distractors are present. Other methods are trained with data masking as per Zhao & Linderman (2023).

Figure 5 shows sample frames from the CIFAR-pendulum dataset.

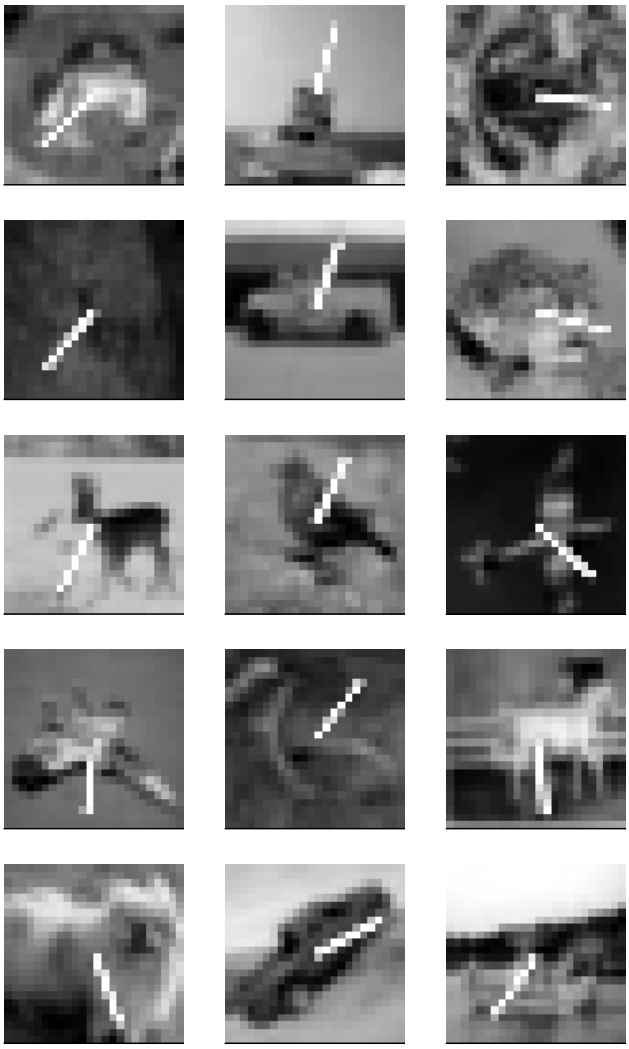


Figure 5: Each column shows frames $0, 10, \ldots, 40$ of a random video sequence from the CIFAR-pendulum dataset.