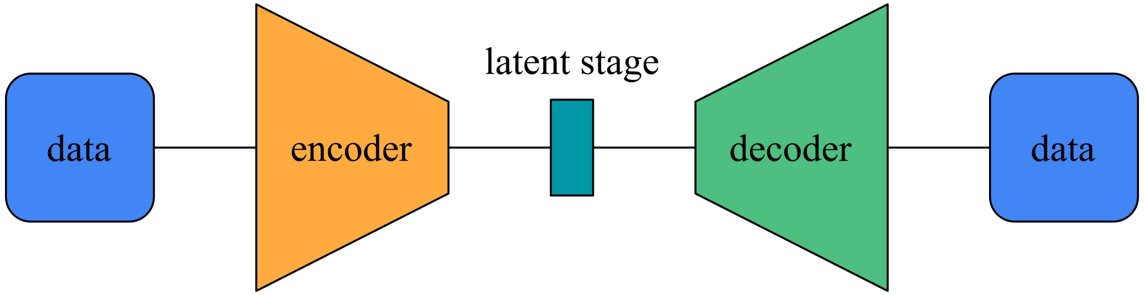
Regularization in latent spaces with high-dimensional linear geometric flows for variational autoencoders

Andrew Gracyk
UIUC, supported by NCSA
Current affiliation, Purdue University mathematics department

Introduction and background

- We present a VAE method with a latent space subject to an intrinsic geometric flow, solved using physics-informed learning, among the latent manifold.
- As a reminder, a VAE is an autoencoder framework, with encoder and decoder neural networks, with an intermediary latent stage. The latent stage is typically regularized with a Gaussian prior, and can be used for generative modeling.

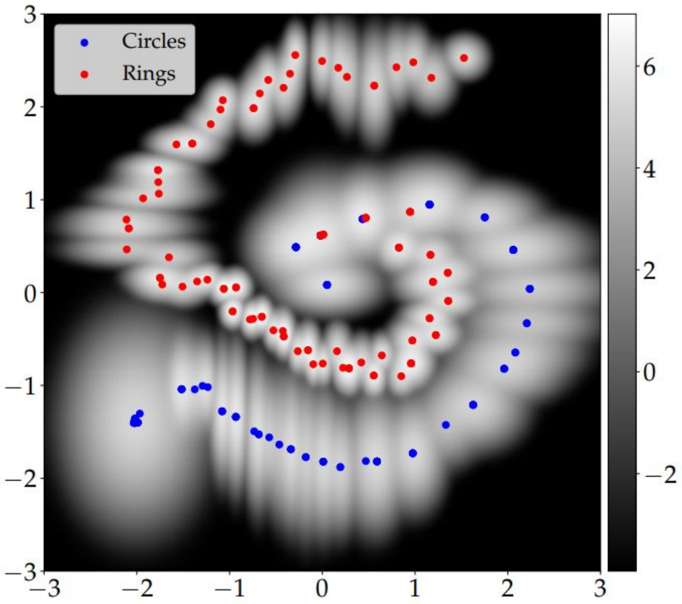


Motivation

- We present a VAE method with a latent space subject to an intrinsic geometric flow, solved using physics-informed learning, among the latent manifold.
- Geometric quality in learning tasks has been characterized by ambient improvement in this process in areas such as:
    - Sampling quality (Chadebec et al.)
    - Representation capacity (learn more complicated data relations better) (Nickel et al.)
    - Identifiability / interpretability (Arvanitidis et al.)
    - Adversarial robustness (**this work**)
- Manifolds in latent spaces have been shown to help learning and robustness (Lopez et al.).
- Can we duplicate this effect of manifold regularization with VAEs but in the setting where the manifold evolves continuously?

VAE latent geometry

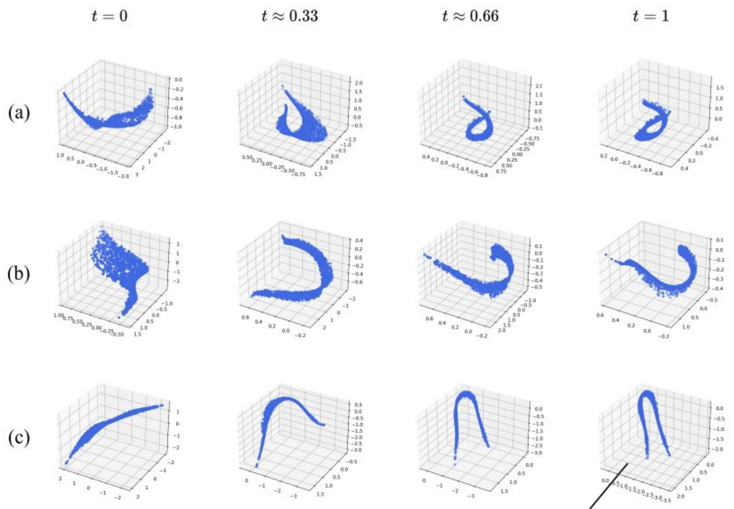
- ➢ Our arguments are less tailored so that geometry itself in latent spaces help learning.
- ➢ Chadebec et al. have demonstrated VAEs can naturally develop structure.
- ➢ Our arguments are more tailored so that we can control the latent geometries to have “sufficiently nice” properties which help learning.



A swiss roll seems to naturally form in the latent means of this data. An intrinsically low manifold is never fully learned due to the variance of the VAE. Figure presented in Chadebec et al., A Geometric Perspective on Variational Autoencoders

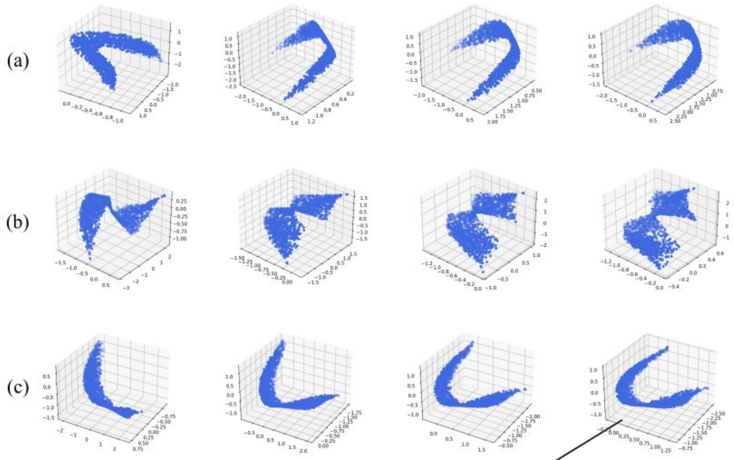
VAE latent geometry

What naturally develops in training with our setup:



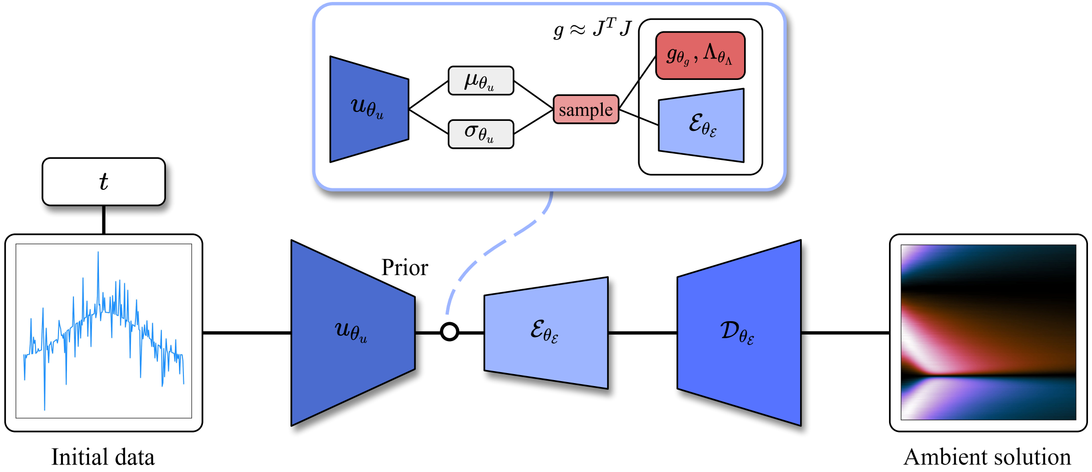
Dissonant, less-structured representations

Our method — latent geometry is significantly more controlled:



Highly controlled, consistent representations

Setup



- ➢ PDE data discretized is mapped to a low-dimensional space (**intrinsic dimension**) where the VAE prior distribution lies.
- ➢ This latent stage is mapped to a subsequent latent stage (with **extrinsic dimension**) where the manifold exists.
- ➢ The manifold point is mapped to PDE data (**discretization length**) at time *t*.

The ELBO for parameterized latent spaces

As in typical in VAEs, our loss is motivated by maximizing log-likelihood $\log p(x_t)$. We will denote $z_t \sim P(\mathcal{M}_t)$ a sampled point from a distribution on the time-evolving manifold. Notice

$$\mathbb{E}_t \log p(x_t) \gtrsim \int_{[0,T]} \log \int_{\mathcal{M}_t} p \frac{q}{q} dV_t dt = \mathbb{E}_{t,z_t} \log \frac{p(x_t, z_t)}{q(z_t, t|x_0)} \quad .$$

Using the change of variables $p(z_t) = p(u)\sqrt{\det((J^{-1})^T J^{-1})}$ (the Jacobian is not square), we get

$$\mathbb{E}_{t,z_t} \log \frac{p}{q} = \mathbb{E}_{t,z_t} \log \frac{p(x_t|z_t)p(z_t)}{q\sqrt{\det((J^{-1})^T J^{-1})}}$$

$$= \mathbb{E}_{t,z_t} \left[\log p(x_t|z_t) + \log \frac{p(u)\sqrt{\det((J^{-1})^T J^{-1})}}{q(u|x_0)\sqrt{\det((J^{-1})^T J^{-1})}} \right]$$

$$\propto -\mathbb{E}_{t,u} \log p(x_t|z_t = \text{encoder}(u,t)) + D_{KL}(q(u|x_0) \parallel p(u))$$

We have noticed the relation between z and u is deterministic. This is our primary loss.

Our geometric flow PDE on the Riemannian metric

We consider a PDE (ODE) on the Riemannian metric

$$\partial_t g(u,t) = -A(u,t)^T A(u,t) g(u,t) + \alpha(-g(u,t) + \Sigma(u))$$

which we will enforce in a loss, which we will develop in a moment. Here, g and A are neural networks. Σ is a fixed Riemannian metric baseline, such as that of the sphere.

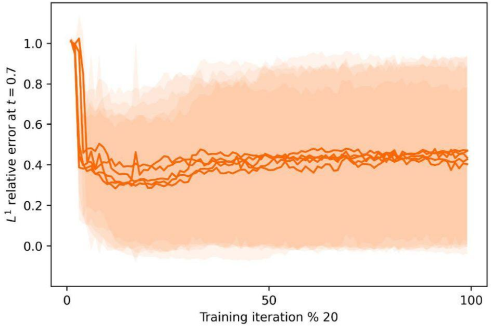
This PDE is linear, efficient to compute because it only requires one time derivative, solved with automatic differentiation.

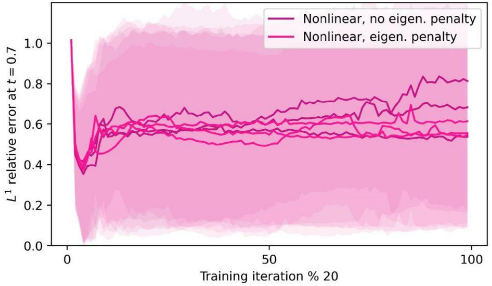
Theorem 1. The geometric flow of equation 22 is a gradient flow with respect to the Euclidean metric and the functional

$$\mathcal{F}(g) = \int_{\mathcal{U}} \frac{1}{2}\langle(\Lambda + \alpha I)g, g\rangle_F - \alpha\langle\Sigma, g\rangle_F du.$$

We will assume nondegeneracy of the metric. This Theorem ensures the metric is nondegenerate, which helps the manifold learn large Lebesgue measure.

Small manifold is less robust



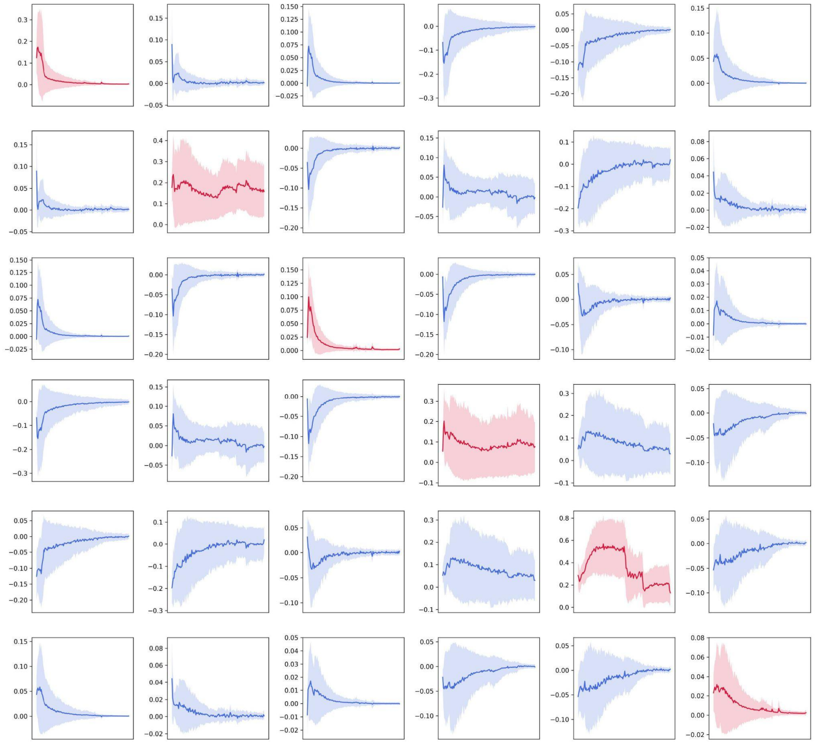


Empirical evidence that our flow (left) outperforms a flow of small Lebesgue measure. The reason a Riemannian metric with smaller values forms on the (right) is because smaller values are easier to satisfy the loss function. We provide evidence the learned metric is small on the next slide.

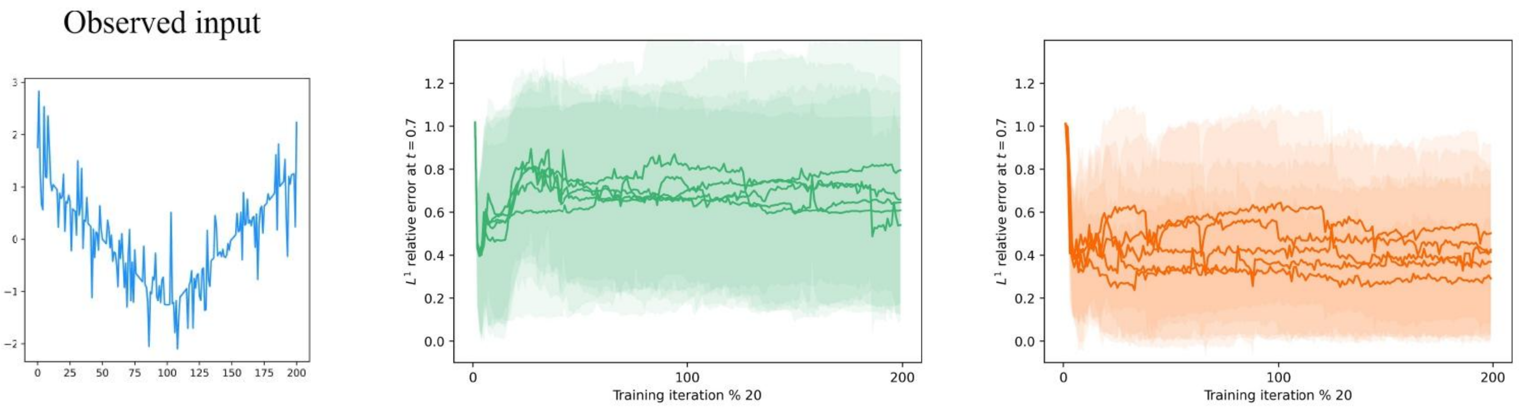
Nondegeneracy is never exactly learned because the decoder will learn distinct latent points for distinct ambient points.

Small metric is learned naturally in training

The Riemannian metric without the steady term is near degenerate. The diagonal of our metric is near zero without this steady term! The off-diagonal follows by Cauchy-Schwarz for an induced metric.



Our method’s robustness



We present evidence that our method promotes adversarial robustness. Our method (orange) achieves lower L1 error across training iteration versus a vanilla VAE (green).

Training

We present our training loss function:

Traditional VAE loss

Regularization term

$$\underbrace{\alpha\mathbb{E}_{t\sim U[0,T],\,u\sim\tilde{q}(\cdot|x_0)}[-\log p(x_t|z_t)] + \beta D_{\mathrm{KL}}(\tilde{q}(u|x_0)||p(u))}_{\text{Our derived modified ELBO}}$$

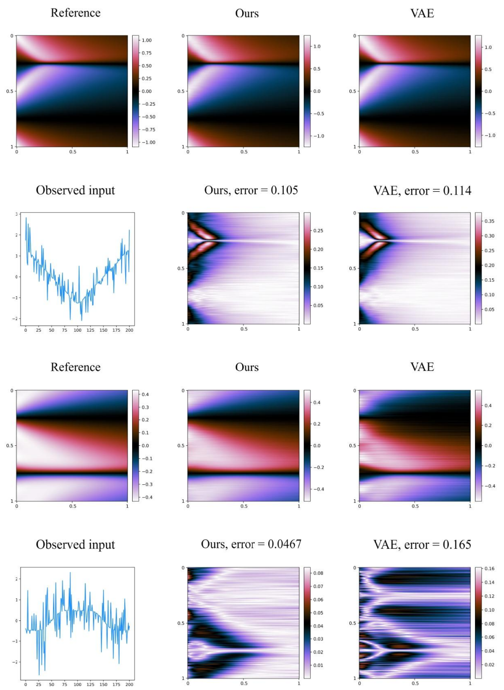
$$+\underbrace{\mathbb{E}_{t\sim U[0,T],\,u\sim\tilde{q}(\cdot|x_0)}\left[\gamma_{\text{geo flow}}\left\|(\partial_t + \Lambda_{\theta_\Lambda}(u,t))g_{\theta_g}(u,t) + \alpha\left(g_{\theta_g}(u,t) - \Sigma(u)\right)\right\|_F^2\right]}_{\text{Physics-informed geometric flow regularization}}$$

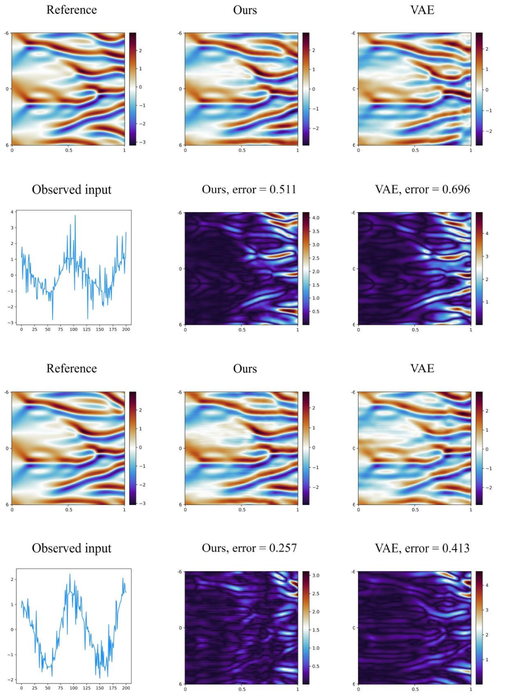
$$+\underbrace{\mathbb{E}_{t\sim U[0,T],\,u\sim\tilde{q}(\cdot|x_0)}\left[\gamma_{\text{metric}}\left\|g_{\theta_g}(u,t) - J^\top J\right\|_F^2\right]}_{\text{Matching induced metric with simulated Riemannian metric}}.$$

Results

Burger's equation					
VAE	$t=0$	$t=0.25$	$t=0.5$	$t=0.75$	$t=1$
In-distribution	3.90e−2±2.26e−2	1.56e−2±1.50e−2	1.59e−2±1.74e−2	1.80e−2±2.00e−2	3.10e−2±3.49e−2
Out-of-distribution 1	2.61e−1±1.73e−1	2.81e−1±3.31e−1	3.32e−1±4.91e−1	3.42e−1±5.61e−1	3.35e−1±5.78e−1
Out-of-distribution 2	2.76e−1±2.22e−1	2.99e−1±3.73e−1	3.22e−1±4.86e−1	3.05e−1±4.50e−1	2.85e−1±4.00e−1
Out-of-distribution 3	1.19e0 ± 9.14e−1	1.20e0 ± 1.49e0	1.12e0 ± 1.46e0	1.00e0 ± 1.22e0	9.07e−1 ± 1.04e0
Out-of-distribution 4	3.05e−1±3.14e−1	3.38e−1±6.65e−1	4.25e−1±1.25e−1	4.64e−1±1.57e−1	4.62e−1 ± 1.66e0
VAE, extended	$t=0$	$t=0.25$	$t=0.5$	$t=0.75$	$t=1$
In-distribution	8.05e−2±5.41e−2	2.83e−2±2.92e−2	2.31e−2±3.19e−2	2.55e−2±4.64e−2	3.65e−2±5.83e−2
Out-of-distribution 1	2.28e−1±1.87e−1	2.37e−1±2.76e−1	2.41e−1±3.14e−1	2.24e−1±3.02e−1	2.08e−1±2.88e−1
Out-of-distribution 2	2.03e−1±1.65e−1	2.13e−1±2.39e−1	2.18e−1±2.92e−1	2.03e−1±2.77e−1	1.89e−1±2.52e−1
Out-of-distribution 3	7.92e−1±7.32e−1	8.84e−1 ± 1.06e0	8.43e−1 ± 1.14e0	7.40e−1 ± 1.00e0	6.53e−1±8.59e−1
Out-of-distribution 4	2.89e−1±3.15e−1	2.99e−1±6.74e−1	3.62e−1 ± 1.23e0	3.94e−1 ± 1.52e0	4.03e−1 ± 1.61e0
VAE-DLM unweighted (ours)	$t=0$	$t=0.25$	$t=0.5$	$t=0.75$	$t=1$
In-distribution	8.62e−2±5.71e−2	3.00e−2±2.97e−2	3.04e−2±5.11e−2	2.94e−2±5.40e−2	4.63e−2±8.23e−2
Out-of-distribution 1	**2.19e−1±1.95e−1**	**1.77e−1±2.08e−1**	**1.71e−1±2.39e−1**	**1.65e−1±2.48e−1**	**1.68e−1±2.59e−1**
Out-of-distribution 2	**1.98e−1±1.49e−1**	**1.76e−1±1.99e−1**	**1.63e−1±2.42e−1**	**1.48e−1±2.32e−1**	**1.42e−1±2.24e−1**
Out-of-distribution 3	7.06e−1±5.54e−1	**6.51e−1±7.37e−1**	**5.38e−1±6.95e−1**	**4.44e−1±5.47e−1**	**4.23e−1±4.03e−1**
Out-of-distribution 4	2.54e−1±2.59e−1	2.40e−1±5.36e−1	2.68e−1±9.36e−1	2.91e−1 ± 1.14e0	3.10e−1 ± 1.25e0
VAE-DLM weighted (ours)	$t=0$	$t=0.25$	$t=0.5$	$t=0.75$	$t=1$
In-distribution	8.74e−2±5.77e−2	2.92e−2±2.54e−2	3.13e−2±4.80e−2	2.82e−2±3.95e−2	5.16e−2±1.08e−1
Out-of-distribution 1	2.25e−1±1.53e−1	1.91e−1±2.31e−1	1.83e−1±2.58e−1	1.84e−1±2.70e−1	1.97e−1±2.88e−1
Out-of-distribution 2	2.01e−1±1.35e−1	1.81e−1±1.92e−1	1.78e−1±2.40e−1	1.71e−1±2.31e−1	1.70e−1±2.20e−1
Out-of-distribution 3	6.42e−1±5.60e−1	6.62e−1±7.59e−1	5.66e−1±6.67e−1	5.07e−1±5.81e−1	5.00e−1±5.59e−1
Out-of-distribution 4	**2.52e−1±2.38e−1**	**2.31e−1±4.69e−1**	**2.58e−1±8.29e−1**	**2.72e−1±9.98e−1**	**2.86e−1 ± 1.07e0**

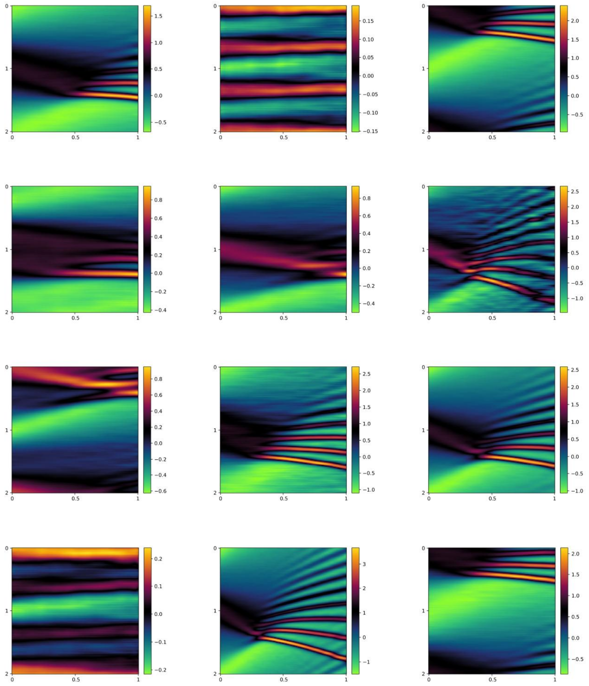
Results


Reference
Ours
VAE
Observed input
Ours, error = 0.105
VAE, error = 0.114
Reference
Ours
VAE
Observed input
Ours, error = 0.0467
VAE, error = 0.165


Reference
Ours
VAE
Observed input
Ours, error = 0.511
VAE, error = 0.696
Reference
Ours
VAE
Observed input
Ours, error = 0.257
VAE, error = 0.413

Generative modeling is also possible with manifold regularization

Here, we sample the prior in local coordinates, which is subsequently mapped to the manifold. The prior will roughly match a Gaussian due to the KL divergence in the loss.



Thanks for listening! Questions?

Key references

- Sifan Wang, Shyam Sankaran, Hanwen Wang, and Paris Perdikaris. An expert's guide to training physics-informed neural networks, 2023.
- Ryan Lopez and Paul J. Atzberger. Gd-vaes: Geometric dynamic variational autoencoders for learning nonlinear dynamics and dimension reductions, 2022.
- Ryan Lopez and Paul J. Atzberger. Variational autoencoders for learning nonlinear dynamics of physical systems, 2021.
- Cheongjae Jang, Yonghyeon Lee, Yung-Kyun Noh, and Frank C. Park. Geometrically regularized autoencoders for non-euclidean data. In The Eleventh International Conference on Learning Representations, 2023.
- Clement Chadebec and Stephanie Allassonniere. A geometric perspective on variational autoencoders, 2022.