Variational Autoencoders for Learning Nonlinear Dynamics of Physical Systems

Ryan Lopez,[3] **Paul J. Atzberger** [1,2,+*]

[1] Department of Mathematics, University of California Santa Barbara (UCSB).
[2] Department of Mechanical Engineering, University of California Santa Barbara (UCSB).
[3] Department of Physics, University of California Santa Barbara (UCSB).
[+] atzberg@gmail.com
http://atzberger.org/

Abstract

We develop data-driven methods for incorporating physical information for priors to learn parsimonious representations of nonlinear systems arising from parameterized PDEs and mechanics. Our approach is based on Variational Autoencoders (VAEs) for learning nonlinear state space models from observations. We develop ways to incorporate geometric and topological priors through general manifold latent space representations. We investigate the performance of our methods for learning low dimensional representations for the nonlinear Burgers equation and constrained mechanical systems.

Introduction

The general problem of learning dynamical models from a time series of observations has a long history spanning many fields [51, 67, 15, 35] including in dynamical systems [8, 67, 68, 47, 50, 52, 32, 19, 23], control [9, 51, 60, 63], statistics [1, 48, 26], and machine learning [15, 35, 46, 58, 3, 73]. Referred to as system identification in control and engineering, many approaches have been developed starting with linear dynamical systems (LDS). These includes the Kalman Filter and extensions [39, 22, 28, 70, 71], Principle Orthogonal Decomposition (POD) [12, 49], and more recently Dynamic Mode Decomposition (DMD) [63, 45, 69] and Koopman Operator approaches [50, 20, 42]. These successful and widely-used approaches rely on assumptions on the model structure, most commonly, that a time-invariant LDS provides a good local approximation or that noise is Gaussian.

There also has been research on more general nonlinear system identification [1, 65, 15, 35, 66, 47, 48, 51]. Nonlinear systems pose many open challenges and fewer unified approaches given the rich behaviors of nonlinear dynamics. For classes of systems and specific application domains, methods have been developed which make different levels of assumptions about the underlying structure of the dynamics. Methods for learning nonlinear dynamics include the NARAX and NOE approaches with function approximators based on neural networks and other models

classes [51, 67], sparse symbolic dictionary methods that are linear-in-parameters such as SINDy [9, 64, 67], and dynamic Bayesian networks (DBNs), such as Hidden Markov Chains (HMMs) and Hidden-Physics Models [58, 54, 62, 5, 43, 26].

A central challenge in learning non-linear dynamics is to obtain representations not only capable of reproducing similar outputs as observed directly in the training dataset but to infer structures that can provide stable more long-term extrapolation capabilities over multiple future steps and input states. In this work, we develop learning methods aiming to obtain robust non-linear models by providing ways to incorporate more structure and information about the underlying system related to smoothness, periodicity, topology, and other constraints. We focus particularly on developing Probabilistic Autoencoders (PAE) that incorporate noise-based regularization and priors to learn lower dimensional representations from observations. This provides the basis of non-linear state space models for prediction. We develop methods for incorporating into such representations geometric and topological information about the system. This facilitates capturing qualitative features of the dynamics to enhance robustness and to aid in interpretability of results. We demonstrate and perform investigations of our methods to obtain models for reductions of parameterized PDEs and for constrained mechanical systems.

Learning Nonlinear Dynamics with Variational Autoencoders (VAEs)

We develop data-driven approaches based on a Variational Autoencoder (VAE) framework [40]. We learn from observation data a set of lower dimensional representations that are used to make predictions for the dynamics. In practice, data can include experimental measurements, large-scale computational simulations, or solutions of complicated dynamical systems for which we seek reduced models. Reductions aid in gaining insights for a class of inputs or physical regimes into the underlying mechanisms generating the observed behaviors. Reduced descriptions are also helpful in many optimization problems in design and in development of controllers [51].

Standard autoencoders can result in encodings that yield unstructured scattered disconnected coding points for system features $\mathbf{z}$. VAEs provide probabilistic encoders and de-

coders where noise provides regularizations that promote more connected encodings, smoother dependence on inputs, and more disentangled feature components [40]. As we shall discuss, we also introduce other regularizations into our methods to help aid in interpretation of the learned latent representations.
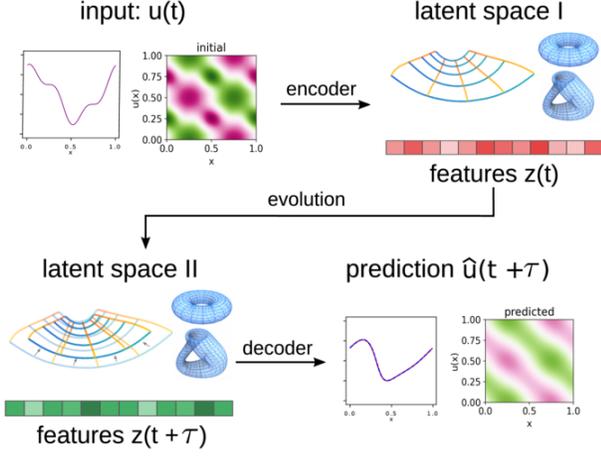
Figure 1: **Learning Nonlinear Dynamics.** Data-driven methods are developed for learning robust models to predict from $u(x,t)$ the non-linear evolution to $u(x,t+\tau)$ for PDEs and other dynamical systems. Probabilistic Autoencoders (PAEs) are utilized to learn representations z of $u(x,t)$ in low dimensional latent spaces with prescribed geometric and topological properties. The model makes predictions using learnable maps that (i) encode an input $u(x,t) \in \mathcal{U}$ as $z(t)$ in latent space *(top)*, (ii) evolve the representation $z(t) \to z(t+\tau)$ *(top-right)*, (iii) decode the representation $z(t+\tau)$ to predict $\hat{u}(x,t+\tau)$ *(bottom-right)*.

We learn VAE predictors using a Maximum Likelihood Estimation (MLE) approach for the Log Likelihood (LL) $\mathcal{L}_{LL} = \log(p_\theta(\mathbf{X}, \mathbf{x}))$. For dynamics of $u(s)$, let $\mathbf{X} = u(t)$ and $\mathbf{x} = u(t+\tau)$. We base p_θ on the autoencoder framework in Figure 1 and 2. We use variational inference to approximate the LL by the Evidence Lower Bound (ELBO) [7] to train a model with parameters θ using encoders and decoders based on minimizing the loss function

$$
\begin{aligned}
\theta^* &= \arg\min_{\theta_e,\theta_d} -\mathcal{L}^B(\theta_e,\theta_d,\theta_\ell;\mathbf{X}^{(i)},\mathbf{x}^{(i)}), \\
\mathcal{L}^B &= \mathcal{L}_{RE} + \mathcal{L}_{KL} + \mathcal{L}_{RR}, \quad\quad (1)\\
\mathcal{L}_{RE} &= E_{\mathfrak{q}_{\theta_e}(\mathbf{z}|\mathbf{X}^{(i)})}\left[\log \mathfrak{p}_{\theta_d}(\mathbf{x}^{(i)}|\mathbf{z}')\right] \\
\mathcal{L}_{KL} &= -\beta\mathcal{D}_{KL}\left(\mathfrak{q}_{\theta_e}(\mathbf{z}|\mathbf{X}^{(i)}) \,\|\, \tilde{\mathfrak{p}}_{\theta_d}(\mathbf{z})\right) \\
\mathcal{L}_{RR} &= \gamma E_{\mathfrak{q}_{\theta_e}(\mathbf{z}'|\mathbf{x}^{(i)})}\left[\log \mathfrak{p}_{\theta_d}(\mathbf{x}^{(i)}|\mathbf{z}')\right].
\end{aligned}
$$

The $\mathfrak{q}_{\theta_e}$ denotes the encoding probability distribution and $\mathfrak{p}_{\theta_d}$ the decoding probability distribution. The loss $\ell = -\mathcal{L}^B$ provides a regularized form of MLE.

The terms $\mathcal{L}_{RE}$ and $\mathcal{L}_{KL}$ arise from the ELBO variational bound $\mathcal{L}_{LL} \geq \mathcal{L}_{RE} + \mathcal{L}_{KL}$ when $\beta = 1$, [7]. This provides a
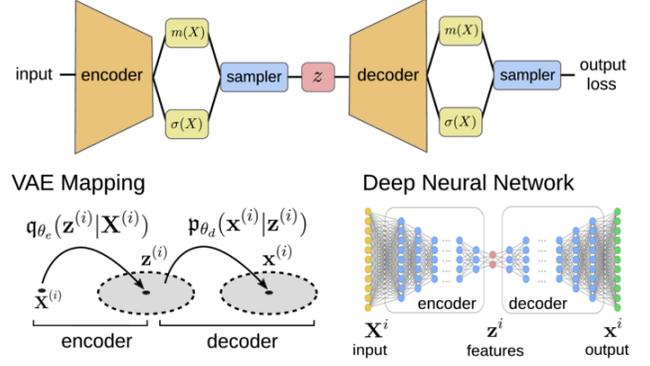
Figure 2: **Variational Autoencoder (VAE).** VAEs [40] are used to learn representations of the nonlinear dynamics. Deep Neural Networks (DNNs) are trained (i) to serve as feature extractors to represent functions $u(x,t)$ and their evolution in a low dimensional latent space as $z(t)$ (encoder $\sim \mathfrak{q}_{\theta_e}$), and (ii) to serve as approximators that can construct predictions $u(x,t+\tau)$ using features $z(t+\tau)$ (decoder $\sim \mathfrak{p}_{\theta_d}$).

way to estimate the log likelihood that the encoder-decoder reproduce the observed data sample pairs $(\mathbf{X}^{(i)}, \mathbf{x}^{(i)})$ using the codes $\mathbf{z}'$ and $\mathbf{z}$. Here, we include a latent-space mapping $\mathbf{z}' = f_{\theta_\ell}(\mathbf{z})$ parameterized by θ_ℓ, which we can use to characterize the evolution of the system or further processing of features. The $\mathbf{X}^{(i)}$ is the input and $\mathbf{x}^{(i)}$ is the output prediction. For the case of dynamical systems, we take $\mathbf{X}^{(i)} \sim u^i(t)$ a sample of the initial state function $u^i(t)$ and the output $\mathbf{x}^{(i)} \sim u^i(t+\tau)$ the predicted state function $u^i(t+\tau)$. We discuss the specific distributions used in more detail below.

The $\mathcal{L}_{KL}$ term involves the Kullback-Leibler Divergence [44, 18] acting similar to a Bayesian prior on latent space to regularize the encoder conditional probability distribution so that for each sample this distribution is similar to p_{θ_d}. We take $p_{\theta_d} = \eta(0, \sigma_0^2)$ a multi-variate Gaussian with independent components. This serves (i) to disentangle the features from each other to promote independence, (ii) provide a reference scale and localization for the encodings $\mathbf{z}$, and (iii) promote parsimonious codes utilizing smaller dimensions than d when possible.

The $\mathcal{L}_{RR}$ term gives a regularization that promotes retaining information in z so the encoder-decoder pair can reconstruct functions. As we shall discuss, this also promotes organization of the latent space for consistency over multi-step predictions and aids in model interpretability.

We use for the specific encoder probability distributions conditional Gaussians $\mathbf{z} \sim \mathfrak{q}_{\theta_e}(\mathbf{z}|\mathbf{x}^{(i)}) = \mathfrak{a}(\mathbf{X}^{(i)}, \mathbf{x}^{(i)}) + \eta(0, \sigma_e^2)$ where η is a Gaussian with variance σ_e^2, (i.e. $\mathbb{E}^{\mathbf{X}^i}[z] = \mathfrak{a}$, $\text{Var}^{\mathbf{X}^i}[z] = \sigma_e^2$). One can think of the learned mean function $\mathfrak{a}$ in the VAE as corresponding to a typical encoder $\mathfrak{a}(\mathbf{X}^{(i)}, \mathbf{x}^{(i)}; \theta_e) = \mathfrak{a}(\mathbf{X}^{(i)}; \theta_e) = \mathbf{z}^{(i)}$ and the variance function $\sigma_e^2 = \sigma_e^2(\theta_e)$ as providing control of a

noise source to further regularize the encoding. Among other properties, this promotes connectedness of the ensemble of latent space codes. For the VAE decoder distribution, we take $\mathbf{x} \sim \mathfrak{p}_{\theta_d}(\mathbf{x}|\mathbf{z}^{(i)}) = \mathfrak{b}(\mathbf{z}^{(i)}) + \eta(0, \sigma_d^2)$. The learned mean function $\mathfrak{b}(\mathbf{z}^{(i)}; \theta_e)$ corresponds to a typical decoder and the variance function $\sigma_e^2 = \sigma_e^2(\theta_d)$ controls the source of regularizing noise.

The terms to be learned in the VAE framework are $(\mathfrak{a}, \sigma_e, f_{\theta_\ell}, \mathfrak{b}, \sigma_d)$ which are parameterized by $\theta = (\theta_e, \theta_d, \theta_\ell)$. In practice, it is useful to treat variances $\sigma_{(\cdot)}$ initially as hyper-parameters. We learn predictors for the dynamics by training over samples of evolution pairs $\{(u_n^i, u_{n+1}^i)\}_{i=1}^m$, where i denotes the sample index and $u_n^i = u^i(t_n)$ with $t_n = t_0 + n\tau$ for a time-scale τ.

To make predictions, the learned models use the following stages: (i) extract from $u(t)$ the features $z(t)$, (ii) evolve $z(t) \to z(t + \tau)$, (iii) predict using $z(t + \tau)$ the $\hat{u}(t + \tau)$, summarized in Figure 1. By composition of the latent evolution map the model makes multi-step predictions of the dynamics.

Learning with Manifold Latent Spaces Roles of Non-Euclidean Geometry and Topology

For many systems, parsimonious representations can be obtained by working with non-euclidean manifold latent spaces, such as a torus for doubly periodic systems or even non-orientable manifolds, such a klein bottle as arises in imaging and perception studies [10]. For this purpose, we learn encoders $\mathcal{E}$ over a family of mappings to a prescribed manifold $\mathcal{M}$ of the form

$$\mathbf{z} = \mathcal{E}_\phi(\mathbf{x}) = \Lambda(\tilde{\mathcal{E}}_\phi(\mathbf{x})) = \Lambda(\mathbf{w}), \quad \mathbf{w} = \tilde{\mathcal{E}}_\phi(\mathbf{x}).$$

We take the map $\tilde{\mathcal{E}}_\phi(\mathbf{x}) : \mathbf{x} \to \mathbf{w}$, where we represent a smooth closed manifold $\mathcal{M}$ of dimension m in $\mathbb{R}^{2m}$, as supported by the Whitney Embedding Theorem [72]. The Λ maps (projects) points $\mathbf{w} \in \mathbb{R}^{2m}$ to the manifold representation $\mathbf{z} \in \mathcal{M} \subset \mathbb{R}^{2m}$. In practice, we accomplish this two ways: (i) we provide an analytic mapping Λ to $\mathcal{M}$, (ii) we provide a high resolution point-cloud representation of the target manifold along with local gradients and use for Λ a quantized mapping to the nearest point on $\mathcal{M}$. We provide more details in Appendix A.

This allows us to learn VAEs with latent spaces for $\mathbf{z}$ with general specified topologies and controllable geometric structures. The topologies of sphere, torus, klein bottle are intrinsically different than $\mathbb{R}^n$. This allows for new types of priors such as uniform on compact manifolds or distributions with more symmetry. As we shall discuss, additional latent space structure also helps in learning more robust representations less sensitive to noise since we can unburden the encoder and decoder from having to learn the embedding geometry and avoid the potential for them making erroneous use of extra latent space dimensions. We also have statistical gains since the decoder now only needs to learn a mapping from the manifold $\mathcal{M}$ for reconstructions of $\mathbf{x}$. These more parsimonious representations also aid identifiability and interpretability of models.

Related Work

Many variants of autoencoders have been developed for making predictions of sequential data, including those based on Recurrent Neural Networks (RNNs) with LSTMs and GRUs [34, 29, 16]. While RNNs provide a rich approximation class for sequential data, they pose for dynamical systems challenges for interpretability and for training to obtain predictions stable over many steps with robustness against noise in the training dataset. Autoencoders have also been combined with symbolic dictionary learning for latent dynamics in [11] providing some advantages for interpretability and robustness, but require specification in advance of a sufficiently expressive dictionary. Neural networks incorporating physical information have also been developed that impose stability conditions during training [53, 46, 24]. The work of [17] investigates combining RNNs with VAEs to obtain more robust models for sequential data and considered tasks related to processing speech and handwriting.

In our work we learn dynamical models making use of VAEs to obtain probabilistic encoders and decoders between euclidean and non-euclidean latent spaces to provide additional regularizations to help promote parsimoniousness, disentanglement of features, robustness, and interpretability. Prior VAE methods used for dynamical systems include [31, 55, 27, 13, 55, 59]. These works use primarily euclidean latent spaces and consider applications including human motion capture and ODE systems. Approaches for incorporating topological information into latent variable representations include the early works by Kohonen on Self-Organizing Maps (SOMs) [41] and Bishop on Generative Topographical Maps (GTMs) based on density networks providing a generative approach [6]. More recently, VAE methods using non-euclidean latent spaces include [37, 38, 25, 14, 21, 2]. These incorporate the role of geometry by augmenting the prior distribution $\tilde{\mathfrak{p}}_{\theta_d}(z)$ on latent space to bias toward a manifold. In the recent work [57], an explicit projection procedure is introduced, but in the special case of a few manifolds having an analytic projection map.

In our work we develop further methods for more general latent space representations, including non-orientable manifolds, and applications to parameterized PDEs and constrained mechanical systems. We introduce more general methods for non-euclidean latent spaces in terms of point-cloud representations of the manifold along with local gradient information that can be utilized within general back-propagation frameworks, see Appendix A. This also allows for the case of manifolds that are non-orientable and having complex shapes. Our methods provide flexible ways to design and control both the topology and the geometry of the latent space by merging or subtracting shapes or stretching and contracting regions. We also consider additional types of regularizations for learning dynamical models facilitating multi-step predictions and more interpretable state space models. In our work, we also consider reduced models for non-linear PDEs, such as Burgers Equations, and learning representations for more general constrained mechanical systems. We also investigate the role of non-linearities making comparisons with other data-driven models.

Results

Burgers' Equation of Fluid Mechanics: Learning Nonlinear PDE Dynamics

We consider the nonlinear viscous Burgers' equation

$$u_t = -uu_x + \nu u_{xx}, \qquad (2)$$

where ν is the viscosity [4, 36]. We consider periodic boundary conditions on $\Omega = [0, 1]$. Burgers equation is motivated as a mechanistic model for the fluid mechanics of advective transport and shocks, and serves as a widely used benchmark for analysis and computational methods.

The nonlinear Cole-Hopf Transform $\mathcal{CH}$ can be used to relate Burgers equation to the linear Diffusion equation $\phi_t = \nu\phi_{xx}$ [36]. This provides a representation of the solution u

$$\phi(x, t) = \mathcal{CH}[u] = \exp\left(-\frac{1}{2\nu}\int_0^x u(x', t)dx'\right)$$

$$u(x, t) = \mathcal{CH}^{-1}[\phi] = -2\nu\frac{\partial}{\partial x}\ln\phi(x, t). \qquad (3)$$

This can be represented by the Fourier expansion

$$\phi(x, t) = \sum_{k=-\infty}^{\infty} \hat{\phi}_k(0)\exp(-4\pi^2 k^2 \nu t) \cdot \exp(i2\pi kx).$$

The $\hat{\phi}_k(0) = \mathcal{F}_k[\phi(x, 0)]$ and $\phi(x, t) = \mathcal{F}^{-1}[\{\hat{\phi}_k(0)\exp(-4\pi^2 k^2 \nu t)\}]$ with $\mathcal{F}$ the Fourier transform. This provides an analytic representation of the solution of the viscous Burgers equation $u(x, t) = \mathcal{CH}^{-1}[\phi(x, t)]$ where $\hat{\phi}(0) = \mathcal{F}[\mathcal{CH}[u(x, 0)]]$. In general, for nonlinear PDEs with initial conditions within a class of functions $\mathcal{U}$, we aim to learn models that provide predictions $u(t + \tau) = \mathcal{S}_\tau u(t)$ approximating the evolution operator $\mathcal{S}_\tau$ over time-scale τ. For the Burgers equation, the $\mathcal{CH}$ provides an analytic way to obtain a reduced order model by truncating the Fourier expansion to $|k| \leq n_f/2$. This provides for the Burgers equation a benchmark model against which to compare our learned models. For general PDEs comparable analytic representations are not usually available, motivating development of data-driven approaches.

We develop VAE methods for learning reduced order models for the responses of nonlinear Burgers Equation when the initial conditions are from a collection of functions $\mathcal{U}$. We learn VAE models that extract from $u(x, t)$ latent variables $z(t)$ to predict $u(x, t + \tau)$. Given the non-uniqueness of representations and to promote interpretability of the model, we introduce the inductive bias that the evolution dynamics in latent space for z is linear of the form $\dot{z} = -\lambda_0 z$, giving exponential decay rate λ_0. For discrete times, we take $z_{n+1} = f_{\theta_\ell}(z_n) = \exp(-\lambda_0\tau) \cdot z_n$, where $\theta_\ell = (\lambda_0)$. We still consider general nonlinear mappings for the encoders and decoders which are represented by deep neural networks. We train the model on the pairs $(u(x, t), u(x, t + \tau))$ by drawing m samples of $u^i(x, t_i) \in \mathcal{S}_{t_i}\mathcal{U}$ which generates the evolved state under Burgers equation $u^i(x, t_i + \tau)$ over time-scale τ. We perform VAE studies with parameters $\nu = 2 \times 10^{-2}$, $\tau = 2.5 \times 10^{-1}$ with VAE

Deep Neural Networks (DNNs) with layer sizes (in)-400-400-(out), ReLU activations, and $\gamma = 0.5$, $\beta = 1$, and initial standard deviations $\sigma_d = \sigma_e = 4 \times 10^{-3}$. We show results of our VAE model predictions in Figure 3 and Table 1.
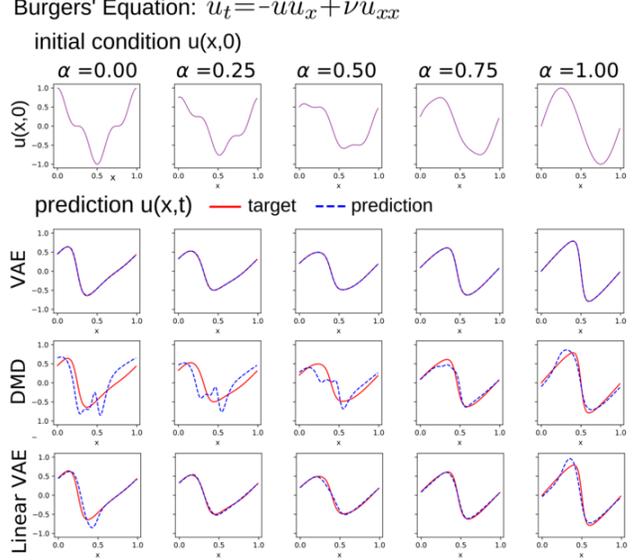
Figure 3: **Burgers' Equation: Prediction of Dynamics.** We consider responses for $\mathcal{U}_1 = \{u \,|\, u(x, t; \alpha) = \alpha\sin(2\pi x) + (1-\alpha)\cos^3(2\pi x)\}$. Predictions are made for the evolution u over the time-scale τ satisfying equation 2 with initial conditions in $\mathcal{U}_1$. We find our nonlinear VAE methods are able to learn with 2 latent dimensions the dynamics with errors $< 1\%$. Methods such as DMD [63, 69] with 3 modes which are only able to use a single linear space to approximate the initial conditions and prediction encounter challenges in approximating the nonlinear evolution. We find our linear VAE method with 2 modes provides some improvements, by allowing for using different linear spaces for representing the input and output functions, but at the cost of additional computations. Results are summarized in Table 1.

We show the importance of the non-linear approximation properties of our VAE methods in capturing system behaviors by making comparisons with Dynamic Mode Decomposition (DMD) [63, 69], Principle Orthogonal Decomposition (POD) [12], and a linear variant of our VAE approach. Recent CNN-AEs have also studied related advantages of nonlinear approximations [46]. Some distinctions in our work is the use of VAEs to further regularize AEs and using topological latent spaces to facilitate further capturing of structure. The DMD and POD are widely used and successful approaches that aim to find an optimal linear space on which to project the dynamics and learn a linear evolution law for system behaviors. DMD and POD have been successful in obtaining models for many applications, including steady-state fluid mechanics and transport problems [69, 63]. However, given their inherent linear approximations they can encounter well-known challenges related to translational and rotational invariances, as arise in advective phenomena and other settings [8]. Our comparison studies can be found in
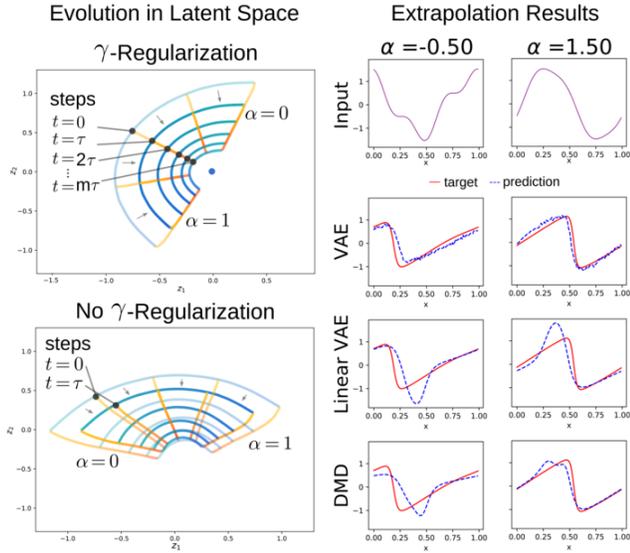
Figure 4: **Burgers' Equation: Latent Space Representations and Extrapolation Predictions.** We show the latent space representation z of the dynamics for the input functions $u(\cdot, t; \alpha) \in \mathcal{U}_1$. VAE organizes for u the learned representations $z(\alpha, t)$ in parameter α *(blue-green)* into circular arcs that are concentric in the time parameter t, *(yellow-orange) (left)*. The reconstruction regularization with γ aligns subsequent time-steps of the dynamics in latent space facilitating multi-step predictions. The learned VAE model exhibits a level of extrapolation to predict dynamics even for some inputs $u \notin \mathcal{U}_1$ beyond the training dataset *(right)*.

Table 1.

We also considered how our VAE methods performed when adjusting the parameters β for the strength of the prior $\tilde{\mathfrak{p}}$ as in β-VAEs [33] and γ for the strength of the reconstruction regularization. The reconstruction regularization has a significant influence on how the VAE organizes representations in latent space and the accuracy of predictions of the dynamics, especially over multiple steps, see Figure 4 and Table 1. The regularization serves to align representations consistently in latent space facilitating multi-step compositions. We also found our VAE learned represenations capable of some level of extrapolation beyond the training dataset. When varying β, we found that larger values improved the multiple step accuracy whereas small values improved the single step accuracy, see Table 1.

Constrained Mechanics: Learning with Non-Euclidean Latent Spaces

To learn more parsimonous and robust representations of physical systems, we develop methods for latent spaces having geometries and topologies more general than euclidean space. This is helpful in capturing inherent structure such as periodicities or other symmetries. We consider physical systems with constrained mechanics, such as the arm mechanism for reaching for objects in figure 5. The observa-

Method	Dim	0.25s	0.50s	0.75s	1.00s
VAE Nonlinear	2	**4.44e-3**	**5.54e-3**	**6.30e-3**	**7.26e-3**
VAE Linear	2	9.79e-2	1.21e-1	1.17e-1	1.23e-1
DMD	3	2.21e-1	1.79e-1	1.56e-1	1.49e-1
POD	3	3.24e-1	4.28e-1	4.87e-1	5.41e-1
Cole-Hopf-2	2	5.18e-1	4.17e-1	3.40e-1	1.33e-1
Cole-Hopf-4	4	5.78e-1	6.33e-2	9.14e-3	1.58e-3
Cole-Hopf-6	6	1.48e-1	2.55e-3	9.25e-5	7.47e-6

γ	0.00s	0.25s	0.50s	0.75s	1.00s
0.00	1.600e-01	6.906e-03	1.715e-01	3.566e-01	5.551e-01
0.50	1.383e-02	1.209e-02	1.013e-02	9.756e-03	1.070e-02
2.00	1.337e-02	1.303e-02	9.202e-03	8.878e-03	1.118e-02

β	0.00s	0.25s	0.50s	0.75s	1.00s
0.00	1.292e-02	1.173e-02	1.073e-02	1.062e-02	1.114e-02
0.50	1.190e-02	1.126e-02	1.072e-02	1.153e-02	1.274e-02
1.00	1.289e-02	1.193e-02	7.903e-03	7.883e-03	9.705e-03
4.00	1.836e-02	1.677e-02	8.987e-03	8.395e-03	8.894e-03

Table 1: **Burgers' Equation: Prediction Accuracy.** The reconstruction L^1-relative errors in predicting $u(x, t)$ for our VAE methods, Dynamic Model Decomposition (DMD), and Principle Orthogonal Decomposition (POD), and reduction by Cole-Hopf (CH), over multiple-steps and number of latent dimensions (Dim) *(top)*. Results when varying the strength of the reconstruction regularization γ and prior β *(bottom)*.

tions are taken to be the two locations $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^2$ giving $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{R}^4$. When the segments are rigidly constrained these configurations lie on a manifold (torus). We can also allow the segments to extend and consider more exotic constraints such as the two points $\mathbf{x}_1, \mathbf{x}_2$ must be on a klein bottle in $\mathbb{R}^4$. Related situations arise in other areas of imaging and mechanics, such as in pose estimation and in studies of visual perception [56, 10, 61]. For the arm mechanics, we can use this prior knowledge to construct a torus latent space represented by the product space of two circles $S^1 \times S^1$. To obtain a learnable class of manifold encoders, we use the family of maps $\mathcal{E}_\theta = \Lambda(\tilde{\mathcal{E}}_\theta(x))$, with $\tilde{\mathcal{E}}_\theta(x)$ into $\mathbb{R}^4$ and $\Lambda(\mathbf{w}) = \Lambda(w_1, w_2, w_3, w_4) = (z_1, z_2, z_3, z_4) = \mathbf{z}$, where $(z_1, z_2) = (w_1, w_2)/\|(w_1, w_2)\|$, $(z_3, z_4) = (w_3, w_4)/\|(w_3, w_4)\|$, see VAE Section and Appendix A. For the case of klein bottle constraints, we use our point-cloud representation of the non-orientable manifold with the parameterized embedding in $\mathbb{R}^4$

$$z_1 = (a + b\cos(u_2))\cos(u_1) \quad z_2 = (a + b\cos(u_2))\sin(u_1)$$
$$z_3 = b\sin(u_2)\cos\left(\tfrac{u_1}{2}\right) \quad z_4 = b\sin(u_2)\sin\left(\tfrac{u_1}{2}\right),$$

with $u_1, u_2 \in [0, 2\pi]$. The $\Lambda(\mathbf{w})$ is taken to be the map to the nearest point of the manifold $\mathcal{M}$, which we compute numerically along with the needed gradients for backpropagation as discussed in Appendix A.

Our VAE methods are trained with encoder and decoder DNN's having layers of sizes (in)-100-500-100-(out) with Leaky-ReLU activations with s = 1e-6 with results reported in Figure 5 and Table 2. We find learning representations is improved by use of the manifold latent spaces, in these trials even showing a slight edge over $\mathbb{R}^4$. When the wrong
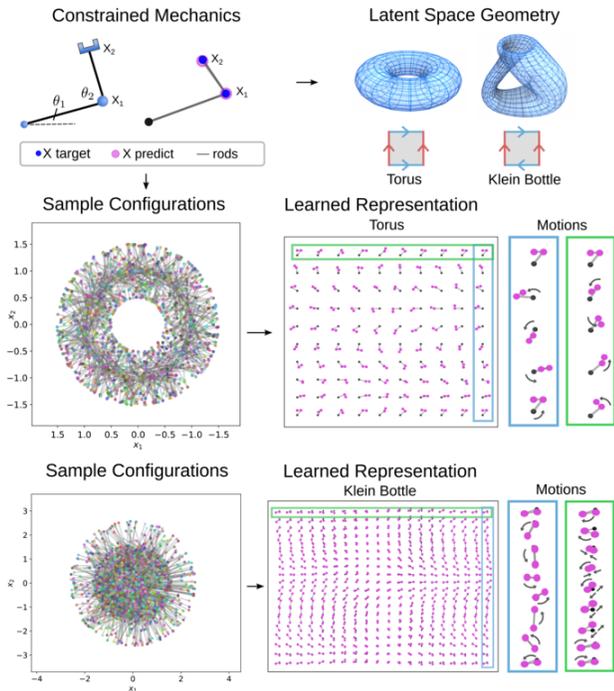
Figure 5: **VAE Representations of Motions using Manifold Latent Spaces.** We learn from observations representations for constrained mechanical systems using general non-euclidean manifolds latent spaces $\mathcal{M}$. The arm mechanism has configurations $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2) \in \mathbb{R}^4$. For rigid segments, the motions are constrained to be on a manifold (torus) $\mathcal{M} \subset \mathbb{R}^4$. For extendable segments, we can also consider more exotic constraints, such as requiring $\mathbf{x}_1, \mathbf{x}_2$ to be on a klein bottle in $\mathbb{R}^4$ *(top)*. Results of our VAE methods for learned representations for motions under these constraints are shown. VAE learns the segment length constraint and two nearly decoupled coordinates for the torus dataset that mimic the roles of angles. VAE learns for the klein bottle dataset two segment motions to generate configurations *(middle and bottom)*.

Torus	epoch			
method	1000	2000	3000	final
VAE 2-Manifold	**6.6087e-02**	**6.6564e-02**	**6.6465e-02**	**6.6015e-02**
VAE $\mathbb{R}^2$	1.6540e-01	1.2931e-01	9.9903e-02	8.0648e-02
VAE $\mathbb{R}^4$	8.0006e-02	7.6302e-02	7.5875e-02	7.5626e-02
VAE $\mathbb{R}^{10}$	8.3411e-02	8.4569e-02	8.4673e-02	8.4143e-02
with noise σ	0.01	0.05	0.1	0.5
VAE 2-Manifold	**6.7099e-02**	**8.0608e-02**	**1.1198e-01**	**4.1988e-01**
VAE $\mathbb{R}^2$	8.5879e-02	9.7220e-02	1.2867e-01	4.5063e-01
VAE $\mathbb{R}^4$	7.6347e-02	9.0536e-02	1.2649e-01	4.9187e-01
VAE $\mathbb{R}^{10}$	8.4780e-02	1.0094e-01	1.3946e-01	5.2050e-01
Klein Bottle	epoch			
method	1000	2000	3000	final
VAE 2-Manifold	**5.7734e-02**	**5.7559e-02**	**5.7469e-02**	**5.7435e-02**
VAE $\mathbb{R}^2$	1.1802e-01	9.0728e-02	8.0578e-02	7.1026e-02
VAE $\mathbb{R}^4$	6.9057e-02	6.5593e-02	6.4047e-02	6.3771e-02
VAE $\mathbb{R}^{10}$	6.8899e-02	6.9802e-02	7.0953e-02	6.8871e-02
with noise σ	0.01	0.05	0.1	0.5
VAE 2-Manifold	**5.9816e-02**	**6.9934e-02**	**9.6493e-02**	**4.0121e-01**
VAE $\mathbb{R}^2$	1.0120e-01	1.0932e-01	1.3154e-01	4.8837e-01
VAE $\mathbb{R}^4$	6.3885e-02	7.6096e-02	1.0354e-01	4.5769e-01
VAE $\mathbb{R}^{10}$	7.4587e-02	8.8233e-02	1.2082e-01	4.8182e-01

Table 2: **Manifold Latent Variable Model: VAE Reconstruction Errors** The L^2-relative errors of reconstruction for our VAE methods. The final is the lowest value during training. The manifold latent spaces show improved learning. When an incompatible topology is used, such as $\mathbb{R}^2$, this can result in deterioration in learned representations. With noise in the input $\tilde{X} = X + \sigma\eta(0, 1)$ and reconstructing the target X, the manifold latent spaces also show improvements for learning.

topology is used, such as in $\mathbb{R}^2$, we find in both cases a significant deterioration in the reconstruction accuracy, see Table 2. This arises since the encoder must be continuous and hedge against the noise regularizations. This results in an incurred penalty for a subset of configurations. The encoder exhibits non-injectivity and a rapid doubling back over the space to accommodate the decoder by lining up nearby configurations in the topology of the input space manifold to handle noise perturbations in z from the probabilistic nature of the encoding. We also studied robustness when training with noise for $\tilde{X} = X + \sigma\eta(0, 1)$ and measuring accuracy for reconstruction relative to target X. As the noise increases, we see that the manifold latent spaces improve reconstruction accuracy acting as a filter through restricting the representation. The probabilistic decoder will tend to learn to estimate the mean over samples of a common underlying configuration and with the manifold latent space restrictions is more likely to use a common latent representation. For $\mathbb{R}^d$ with $d > 2$, the extraneous dimensions in the latent space can result in overfitting of the encoder to the noise. We see as d becomes larger the reconstruction accuracy decreases, see Table 2. These results demonstrate how geometric priors can aid learning in constrained mechanical systems.

Conclusions

We developed VAE's for learning robustly nonlinear dynamics of physical systems by introducing methods for latent representations utilizing general geometric and topological structures. We demonstrated our methods for learning the non-linear dynamics of PDEs and constrained mechanical systems. We expect our methods can also be used in other physics-related tasks and problems to leverage prior geometric and topological knowledge for improving learning for nonlinear systems.

Acknowledgments

References

[1] Archer, E.; Park, I. M.; Buesing, L.; Cunningham, J.; and Paninski, L. 2015. Black box variational inference for state space models. *arXiv preprint arXiv:1511.07367* URL https://arxiv.org/abs/1511.07367.

[2] Arvanitidis, G.; Hansen, L. K.; and Hauberg, S. 2018. Latent Space Oddity: on the Curvature of Deep Generative Models. In *International Conference on Learning Representations*. URL https://openreview.net/forum?id=SJzRZ-WCZ.

[3] Azencot, O.; Yin, W.; and Bertozzi, A. 2019. Consistent dynamic mode decomposition. *SIAM Journal on Applied Dynamical Systems* 18(3): 1565–1585. URL https://www.math.ucla.edu/~bertozzi/papers/CDMD_SIADS.pdf.

[4] Bateman, H. 1915. Some Recent Researches on the Motion of Fluids. *Monthly Weather Review* 43(4): 163. doi:10.1175/1520-0493(1915)43⟨163:SRROTM⟩2.0.CO;2.

[5] Baum, L. E.; and Petrie, T. 1966. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *Ann. Math. Statist.* 37(6): 1554–1563. doi:10.1214/aoms/1177699147. URL https://doi.org/10.1214/aoms/1177699147.

[6] Bishop, C. M.; Svensén, M.; and Williams, C. K. I. 1996. GTM: A Principled Alternative to the Self-Organizing Map. In Mozer, M.; Jordan, M. I.; and Petsche, T., eds., *Advances in Neural Information Processing Systems 9, NIPS, Denver, CO, USA, December 2-5, 1996*, 354–360. MIT Press. URL http://papers.nips.cc/paper/1207-gtm-a-principled-alternative-to-the-self-organizing-map.

[7] Blei, D. M.; Kucukelbir, A.; and McAuliffe, J. D. 2017. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association* 112(518): 859–877. doi:10.1080/01621459.2017.1285773. URL https://doi.org/10.1080/01621459.2017.1285773.

[8] Brunton, S. L.; and Kutz, J. N. 2019. *Reduced Order Models (ROMs)*, 375–402. Cambridge University Press. doi:10.1017/9781108380690.012.

[9] Brunton, S. L.; Proctor, J. L.; and Kutz, J. N. 2016. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences* 113(15): 3932–3937. ISSN 0027-8424. doi:10.1073/pnas.1517384113. URL https://www.pnas.org/content/113/15/3932.

[10] Carlsson, G.; Ishkhanov, T.; de Silva, V.; and Zomorodian, A. 2008. On the Local Behavior of Spaces of Natural Images. *International Journal of Computer Vision* 76(1): 1–12. ISSN 1573-1405. URL https://doi.org/10.1007/s11263-007-0056-x.

[11] Champion, K.; Lusch, B.; Kutz, J. N.; and Brunton, S. L. 2019. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences* 116(45): 22445–22451. ISSN 0027-8424. doi:10.1073/pnas.1906995116. URL https://www.pnas.org/content/116/45/22445.

[12] Chatterjee, A. 2000. An introduction to the proper orthogonal decomposition. *Current Science* 78(7): 808–817. ISSN 00113891. URL http://www.jstor.org/stable/24103957.

[13] Chen, N.; Karl, M.; and Van Der Smagt, P. 2016. Dynamic movement primitives in latent space of time-dependent variational autoencoders. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 629–636. IEEE. URL https://ieeexplore.ieee.org/document/7803340.

[14] Chen, N.; Klushyn, A.; Ferroni, F.; Bayer, J.; and Van Der Smagt, P. 2020. Learning Flat Latent Manifolds with VAEs. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 1587–1596. Virtual: PMLR. URL http://proceedings.mlr.press/v119/chen20i.html.

[15] Chiuso, A.; and Pillonetto, G. 2019. System Identification: A Machine Learning Perspective. *Annual Review of Control, Robotics, and Autonomous Systems* 2(1): 281–304. doi:10.1146/annurev-control-053018-023744. URL https://doi.org/10.1146/annurev-control-053018-023744.

[16] Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1724–1734. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1179. URL https://www.aclweb.org/anthology/D14-1179.

[17] Chung, J.; Kastner, K.; Dinh, L.; Goel, K.; Courville, A. C.; and Bengio, Y. 2015. A Recurrent Latent Variable Model for Sequential Data. *Advances in neural information processing systems* abs/1506.02216. URL http://arxiv.org/abs/1506.02216.

[18] Cover, T. M.; and Thomas, J. A. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience. ISBN 0471241954.

[19] Crutchfield, J.; and McNamara, B. S. 1987. Equations of Motion from a Data Series. *Complex Syst.* 1.

[20] Das, S.; and Giannakis, D. 2019. Delay-Coordinate Maps and the Spectra of Koopman Operators 175: 1107–1145. ISSN 0022-4715. doi:10.1007/s10955-019-02272-w.

[21] Davidson, T. R.; Falorsi, L.; Cao, N. D.; Kipf, T.; and Tomczak, J. M. 2018. Hyperspherical Variational Auto-Encoders URL https://arxiv.org/abs/1804.00891.

[22] Del Moral, P. 1997. Nonlinear filtering: Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics* 325(6): 653 – 658. ISSN 0764-4442. doi:https://doi.org/10.1016/S0764-4442(97)84778-7. URL http://www.sciencedirect.com/science/article/pii/S0764444297847787.

[23] DeVore, R. A. 2017. *Model Reduction and Approximation: Theory and Algorithms*, chapter Chapter 3: The Theoretical Foundation of Reduced Basis Methods, 137–168. SIAM. doi:10.1137/1.9781611974829.ch3. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611974829.ch3.

[24] Erichson, N. B.; Muehlebach, M.; and Mahoney, M. W. 2019. Physics-informed autoencoders for Lyapunov-stable fluid flow prediction. *arXiv preprint arXiv:1905.10866* .

[25] Falorsi, L.; Haan, P. D.; Davidson, T.; Cao, N. D.; Weiler, M.; Forré, P.; and Cohen, T. 2018. Explorations in Homeomorphic Variational Auto-Encoding. *ArXiv* abs/1807.04689. URL https://arxiv.org/pdf/1807.04689.pdf.

[26] Ghahramani, Z.; and Roweis, S. T. 1998. Learning Nonlinear Dynamical Systems Using an EM Algorithm. In Kearns, M. J.; Solla, S. A.; and Cohn, D. A., eds., *Advances in Neural Information Processing Systems 11, [NIPS Conference, Denver, Colorado, USA, November 30 - December 5, 1998]*, 431–437. The MIT Press. URL http://papers.nips.cc/paper/1594-learning-nonlinear-dynamical-systems-using-an-em-algorithm.

[27] Girin, L.; Leglaive, S.; Bie, X.; Diard, J.; Hueber, T.; and Alameda-Pineda, X. 2020. Dynamical Variational Autoencoders: A Comprehensive Review .

[28] Godsill, S. 2019. Particle Filtering: the First 25 Years and beyond. In *Proc. Speech and Signal Processing (ICASSP) ICASSP 2019 - 2019 IEEE Int. Conf. Acoustics*, 7760–7764.

[29] Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. The MIT Press. ISBN 0262035618. URL https://www.deeplearningbook.org/.

[30] Gross, B.; Trask, N.; Kuberry, P.; and Atzberger, P. 2020. Meshfree methods on manifolds for hydrodynamic flows on curved surfaces: A Generalized Moving Least-Squares (GMLS) approach. *Journal of Computational Physics* 409: 109340. ISSN 0021-9991. doi:https://doi.org/10.1016/j.jcp.2020.109340. URL http://www.sciencedirect.com/science/article/pii/S0021999120301145.

[31] Hernández, C. X.; Wayment-Steele, H. K.; Sultan, M. M.; Husic, B. E.; and Pande, V. S. 2018. Variational encoding of complex dynamics. *Physical Review E* 97(6). ISSN 2470-0053. doi:10.1103/physreve.97.062412. URL http://dx.doi.org/10.1103/PhysRevE.97.062412.

[32] Hesthaven, J. S.; Rozza, G.; and Stamm, B. 2016. Reduced Basis Methods 27–43. ISSN 2191-8198. doi:10.1007/978-3-319-22470-1_3.

[33] Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M. M.; Mohamed, S.; and Lerchner, A. 2017. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *ICLR*. URL https://openreview.net/forum?id=Sy2fzU9gl.

[34] Hochreiter, S.; and Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Comput.* 9(8): 1735–1780. ISSN 0899-7667. doi:10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

[35] Hong, X.; Mitchell, R.; Chen, S.; Harris, C.; Li, K.; and Irwin, G. 2008. Model selection approaches for non-linear system identification: a review. *International Journal of Systems Science* 39(10): 925–946. doi:10.1080/00207720802083018. URL https://doi.org/10.1080/00207720802083018.

[36] Hopf, E. 1950. The partial differential equation $u_t + uu_x = \mu_{xx}$. *Comm. Pure Appl. Math. 3, 201-230* URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160030302.

[37] Jensen, K. T.; Kao, T.-C.; Tripodi, M.; and Hennequin, G. 2020. Manifold GPLVMs for discovering non-Euclidean latent structure in neural data URL https://arxiv.org/abs/2006.07429.

[38] Kalatzis, D.; Eklund, D.; Arvanitidis, G.; and Hauberg, S. 2020. Variational Autoencoders with Riemannian Brownian Motion Priors. *arXiv e-prints* arXiv:2002.05227. URL https://arxiv.org/abs/2002.05227.

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

[40] Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. URL http://arxiv.org/abs/1312.6114.

[41] Kohonen, T. 1982. Self-organized formation of topologically correct feature maps. *Biological cybernetics* 43(1): 59–69. URL https://link.springer.com/article/10.1007/BF00337288.

[42] Korda, M.; Putinar, M.; and Mezić, I. 2020. Data-driven spectral analysis of the Koopman operator. *Applied and Computational Harmonic Analysis* 48(2): 599 – 629. ISSN 1063-5203. doi:https://doi.org/10.1016/j.acha.2018.08.002. URL http://www.sciencedirect.com/science/article/pii/S1063520318300988.

[43] Krishnan, R. G.; Shalit, U.; and Sontag, D. A. 2017. Structured Inference Networks for Nonlinear State Space Models. In Singh, S. P.; and Markovitch, S., eds., *Proceedings of the Thirty-First*

AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA, 2101–2109. AAAI Press. URL http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14215.

[44] Kullback, S.; and Leibler, R. A. 1951. On Information and Sufficiency. *Ann. Math. Statist.* 22(1): 79–86. doi:10.1214/aoms/1177729694. URL https://doi.org/10.1214/aoms/1177729694.

[45] Kutz, J. N.; Brunton, S. L.; Brunton, B. W.; and Proctor, J. L. 2016. *Dynamic Mode Decomposition*. Philadelphia, PA: Society for Industrial and Applied Mathematics. doi:10.1137/1.9781611974508. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611974508.

[46] Lee, K.; and Carlberg, K. T. 2020. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics* 404: 108973. ISSN 0021-9991. doi:https://doi.org/10.1016/j.jcp.2019.108973. URL http://www.sciencedirect.com/science/article/pii/S0021999119306783.

[47] Lusch, B.; Kutz, J. N.; and Brunton, S. L. 2018. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications* 9(1): 4950. ISSN 2041-1723. URL https://doi.org/10.1038/s41467-018-07210-0.

[48] Mania, H.; Jordan, M. I.; and Recht, B. 2020. Active learning for nonlinear system identification with guarantees. *arXiv preprint arXiv:2006.10277* URL https://arxiv.org/pdf/2006.10277.pdf.

[49] Mendez, M. A.; Balabane, M.; and Buchlin, J. M. 2018. Multi-scale proper orthogonal decomposition (mPOD) doi:10.1063/1.5043720.

[50] Mezić, I. 2013. Analysis of Fluid Flows via Spectral Properties of the Koopman Operator. *Annual Review of Fluid Mechanics* 45(1): 357–378. doi:10.1146/annurev-fluid-011212-140652. URL https://doi.org/10.1146/annurev-fluid-011212-140652.

[51] Nelles, O. 2013. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models*. Springer Science & Business Media. URL https://play.google.com/books/reader?id=tyjrCAAAQBAJ&hl=en&pg=GBS.PR3.

[52] Ohlberger, M.; and Rave, S. 2016. Reduced Basis Methods: Success, Limitations and Future Challenges. *Proceedings of the Conference Algoritmy* 1–12. URL http://www.iam.fmph.uniba.sk/amuc/ojs/index.php/algoritmy/article/view/389.

[53] Parish, E. J.; and Carlberg, K. T. 2020. Time-series machine-learning error models for approximate solutions to parameterized dynamical systems. *Computer Methods in Applied Mechanics and Engineering* 365: 112990. ISSN 0045-7825. doi:https://doi.org/10.1016/j.cma.2020.112990. URL http://www.sciencedirect.com/science/article/pii/S0045782520301742.

[54] Pawar, S.; Ahmed, S. E.; San, O.; and Rasheed, A. 2020. Data-driven recovery of hidden physics in reduced order modeling of fluid flows 32: 036602. ISSN 1070-6631. doi:10.1063/5.0002051.

[55] Pearce, M. 2020. The Gaussian Process Prior VAE for Interpretable Latent Dynamics from Pixels. volume 118 of *Proceedings of Machine Learning Research*, 1–12. PMLR. URL http://proceedings.mlr.press/v118/pearce20a.html.

[56] Perea, J. A.; and Carlsson, G. 2014. A Klein-Bottle-Based Dictionary for Texture Representation. *International Journal of Computer Vision* 107(1): 75–97. ISSN 1573-1405. URL https://doi.org/10.1007/s11263-013-0676-2.

[57] Perez Rey, L. A.; Menkovski, V.; and Portegies, J. 2020. Diffusion Variational Autoencoders. In Bessiere, C., ed., *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 2704–2710. International Joint Conferences on Artificial Intelligence Organization. doi:10.24963/ijcai.2020/375. URL https://arxiv.org/pdf/1901.08991.pdf.

[58] Raissi, M.; and Karniadakis, G. E. 2018. Hidden physics models: Machine learning of nonlinear partial differential equations. *Journal of Computational Physics* 357: 125 – 141. ISSN 0021-9991. URL https://arxiv.org/abs/1708.00588.

[59] Roeder, G.; Grant, P. K.; Phillips, A.; Dalchau, N.; and Meeds, E. 2019. Efficient Amortised Bayesian Inference for Hierarchical and Nonlinear Dynamical Systems URL https://arxiv.org/abs/1905.12090.

[60] Samuel H. Rudy, J. Nathan Kutz, S. L. B. 2018. Deep learning of dynamics and signal-noise decomposition with time-stepping constraints. *arXiv:1808:02578* URL https://doi.org/10.1016/j.jcp.2019.06.056.

[61] Sarafianos, N.; Boteanu, B.; Ionescu, B.; and Kakadiaris, I. A. 2016. 3D Human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding* 152: 1 – 20. ISSN 1077-3142. doi:https://doi.org/10.1016/j.cviu.2016.09.002. URL http://www.sciencedirect.com/science/article/pii/S1077314216301369.

[62] Saul, L. K. 2020. A tractable latent variable model for nonlinear dimensionality reduction. *Proceedings of the National Academy of Sciences* 117(27): 15403–15408. ISSN 0027-8424. doi:10.1073/pnas.1916012117. URL https://www.pnas.org/content/117/27/15403.

[63] Schmid, P. J. 2010. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics* 656: 5–28. doi:10.1017/S0022112010001217. URL https://doi.org/10.1017/S0022112010001217.

[64] Schmidt, M.; and Lipson, H. 2009. Distilling Free-Form Natural Laws from Experimental Data 324: 81–85. ISSN 0036-8075. doi:10.1126/science.1165893.

[65] Schoukens, J.; and Ljung, L. 2019. Nonlinear System Identification: A User-Oriented Road Map. *IEEE Control Systems Magazine* 39(6): 28–99. doi:10.1109/MCS.2019.2938121.

[66] Schön, T. B.; Wills, A.; and Ninness, B. 2011. System identification of nonlinear state-space models. *Automatica* 47(1): 39 – 49. ISSN 0005-1098. doi:https://doi.org/10.1016/j.automatica.2010.10.013. URL http://www.sciencedirect.com/science/article/pii/S0005109810004279.

[67] Sjöberg, J.; Zhang, Q.; Ljung, L.; Benveniste, A.; Delyon, B.; Glorennec, P.-Y.; Hjalmarsson, H.; and Juditsky, A. 1995. Nonlinear black-box modeling in system identification: a unified overview. *Automatica* 31(12): 1691 – 1724. ISSN 0005-1098. doi:https://doi.org/10.1016/0005-1098(95)00120-8. URL http://www.sciencedirect.com/science/article/pii/0005109895001208. Trends in System Identification.

[68] Talmon, R.; Mallat, S.; Zaveri, H.; and Coifman, R. R. 2015. Manifold Learning for Latent Variable Inference in Dynamical Systems. *IEEE Transactions on Signal Processing* 63(15): 3843–3856. doi:10.1109/TSP.2015.2432731.

[69] Tu, J. H.; Rowley, C. W.; Luchtenburg, D. M.; Brunton, S. L.; and Kutz, J. N. 2014. On dynamic mode decomposition: Theory and applications. *Journal of Computational Dynamics* URL http://aimsciences.org//article/id/1dfebc20-876d-4da7-8034-7cd3c7ae1161.

[70] Van Der Merwe, R.; Doucet, A.; De Freitas, N.; and Wan, E. 2000. The Unscented Particle Filter. In *Proceedings of the 13th International Conference on Neural Information Processing Systems*, NIPS'00, 563–569. Cambridge, MA, USA: MIT Press.

[71] Wan, E. A.; and Van Der Merwe, R. 2000. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, 153–158. doi:10.1109/ASSPCC.2000.882463.

[72] Whitney, H. 1944. The Self-Intersections of a Smooth n-Manifold in 2n-Space. *Annals of Mathematics* 45(2): 220–246. ISSN 0003486X. URL http://www.jstor.org/stable/1969265.

[73] Yang, Y.; and Perdikaris, P. 2018. Physics-informed deep generative models. *arXiv preprint arXiv:1812.03511* .

Appendix A: Backpropogation of Encoders for Non-Euclidean Latent Spaces given by General Manifolds

We develop methods for using backpropogation to learn encoder maps from $\mathbb{R}^d$ to general manifolds $\mathcal{M}$. We perform learning using the family of manifold encoder maps of the form $\mathcal{E}_\theta = \Lambda(\tilde{\mathcal{E}}_\theta(x))$. This allows for use of latent spaces having general topologies and geometries. We represent the manifold as an embedding $\mathcal{M} \subset \mathbb{R}^{2m}$ and computationally use point-cloud representations along with local gradient information, see Figure 6. To allow for $\mathcal{E}_\theta$ to be learnable, we develop approaches for incorporating our maps into general backpropogation frameworks.
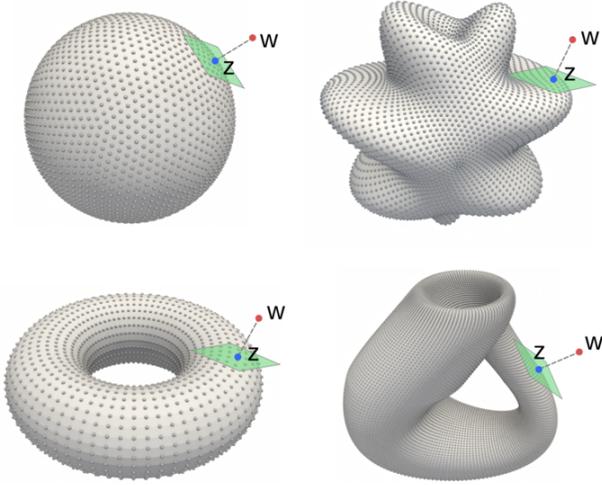
Figure 6: **Learnable Mappings to Manifold Surfaces** We develop methods based on point cloud representations embedded in $\mathbb{R}^n$ for learning latent manifold representations having general geometries and topologies.

For a manifold $\mathcal{M}$ of dimension m, we can represent it by an embedding within $\mathbb{R}^{2m}$, as supported by the Whitney Embedding Theorem [72]. We let $\mathbf{z} = \Lambda(\mathbf{w})$ be a mapping $\mathbf{w} \in \mathbb{R}^{2m}$ to points on the manifold $\mathbf{z} \in \mathcal{M}$. This allows for learning within the family of manifold encoders $w = \tilde{\mathcal{E}}_\theta(x)$ any function from $\mathbb{R}^d$ to $\mathbb{R}^{2m}$. This facilitates use of deep neural networks and other function classes. In practice, we shall take $\mathbf{z} = \Lambda(\mathbf{w})$ to map to the nearest location on the manifold. We can express this as the optimization problem

$$z^* = \arg\min_{z \in \mathcal{M}} \frac{1}{2}\|w - z\|_2^2.$$

We can always express a smooth manifold using local coordinate charts $\sigma^k(u)$, for example, by using a local Monge-Gauge quadratic fit to the point cloud [30]. We can express $z^* = \sigma^k(u^*)$ for some chart k^*. In terms of the coordinate charts $\{\mathcal{U}_k\}$ and local parameterizations $\{\sigma^k(u)\}$ we can express this as

$$u^*, k^* = \arg\min_{k, u \in \mathcal{U}_k} \frac{1}{2}\|w - \sigma^k(u)\|_2^2,$$

where $\Phi_k(u, w) = \frac{1}{2}\|w - \sigma^k(u)\|_2^2$. The w is the input and u^*, k^* is the solution sought. For smooth parameterizations, the optimal solution satisfies

$$G = \nabla_z \Phi_{k^*}(u^*, w) = 0.$$

During learning we need gradients $\nabla_w \Lambda(w) = \nabla_w z$ when w is varied characterizing variations of points on the manifold $z = \Lambda(w)$. We derive these expressions by considering variations $w = w(\gamma)$ for a scalar parameter γ. We can obtain the needed gradients by determining the variations of $u^* = u^*(\gamma)$. We can express these gradients using the Implicit Function Theorem as

$$0 = \frac{d}{d\gamma} G(u^*(\gamma), w(\gamma)) = \nabla_u G \frac{du^*}{d\gamma} + \nabla_w G \frac{dw}{d\gamma}.$$

This implies

$$\frac{du^*}{d\gamma} = -\left[\nabla_u G\right]^{-1} \nabla_w G \frac{dw}{d\gamma}.$$

As long as we can evaluate at u these local gradients $\nabla_u G$, $\nabla_w G$, $dw/d\gamma$, we only need to determine computationally the solution u^*. For the backpropogation framework, we use these to assemble the needed gradients for our manifold encoder maps $\mathcal{E}_\theta = \Lambda(\tilde{\mathcal{E}}_\theta(x))$ as follows.

We first find numerically the closest point in the manifold $z^* \in \mathcal{M}$ and represent it as $z^* = \sigma(u^*) = \sigma^{k^*}(u^*)$ for some chart k^*. In this chart, the gradients can be expressed as

$$G = \nabla_u \Phi(u, w) = -(w - \sigma(u))^T \nabla_u \sigma(u).$$

We take here a column vector convention with $\nabla_u \sigma(u) = [\sigma_{u_1}|\ldots|\sigma_{u_k}]$. We next compute

$$\nabla_u G = \nabla_{uu}\Phi = \nabla_u \sigma^T \nabla_u \sigma - (w - \sigma(u))^T \nabla_{uu}\sigma(u)$$

and

$$\nabla_w G = \nabla_{w,u}\Phi = -I \nabla_u \sigma(u).$$

For implementation it is useful to express this in more detail component-wise as

$$[G]_i = -\sum_k (w_k - \sigma_k(u))\partial_{u_i}\sigma_k(u),$$

with

$$
\begin{aligned}
[\nabla_u G]_{i,j} &= [\nabla_{uu}\Phi]_{i,j} = \sum_k \partial_{u_j}\sigma_k(u)\partial_{u_i}\sigma_k(u) \\
&\quad - \sum_k (w_k - \sigma_k(u))\partial^2_{u_i,u_j}\sigma_k(u) \\
[\nabla_w G]_{i,j} &= [\nabla_{w,u}\Phi]_{i,j} \\
&= -\sum_k \partial_{w_j}w_k\partial_{u_i}\sigma_k(u) = -\partial_{u_i}\sigma_j(u).
\end{aligned}
$$

The final gradient is given by

$$\frac{d\Lambda(w)}{d\gamma} = \frac{dz^*}{d\gamma} = \nabla_u \sigma \frac{du^*}{d\gamma} = -\nabla_u \sigma \left[\nabla_u G\right]^{-1} \nabla_w G \frac{dw}{d\gamma}.$$

In summary, once we determine the point $z^* = \Lambda(w)$ we need only evaluate the above expressions to obtain the needed gradient for learning via backpropogation

$$\nabla_\theta \mathcal{E}_\theta(x) = \nabla_w \Lambda(w)\nabla_\theta \tilde{\mathcal{E}}_\theta(x), \ w = \tilde{\mathcal{E}}_\theta(x).$$

The $\nabla_w \Lambda$ is determined by $d\Lambda(w)/d\gamma$ using $\gamma = w_1, \ldots w_n$. In practice, the $\tilde{\mathcal{E}}_\theta(x)$ is represented by a deep neural network from $\mathbb{R}^d$ to $\mathbb{R}^{2m}$. In this way, we can learn general encoder mappings $\mathcal{E}_\theta(x)$ from $x \in \mathbb{R}^d$ to general manifolds $\mathcal{M}$.