POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

A Multi-Fidelity approach to Deep Kernel Learning of dynamical systems from high-dimensional data sources

Laurea Magistrale in Mathematical Engineering - Statistical Learning

Author: Federico Lancellotti

Advisor: Prof. Andrea Manzoni

Co-advisor: Dr. Nicolò Botteghi

Academic year: 2023-2024

1. Introduction

Data-driven discovery is at the heart of many modern methods we use to model, predict and control dynamical systems. Though data are often high-dimensional, the behaviour exhibited by several complex systems can be effectively captured by a reduced number of latent state variables. Moreover, Multi-Fidelity (MF) methods are often employed to incorporate information from cheaper, but less accurate, sources.
With the present work, we introduce a a data-driven strategy for dimensionality reduction, latent-state model learning and uncertainty quantification, based on a variety of high-dimensional measurements of different degrees of accuracy, generated by dynamical systems whose model is supposed to be unknown, generalizing in a MF sense the framework recently proposed by Botteghi et. al in [1].
In particular, we introduce an ensemble of sub-models, each associated with a level of fidelity and composed of a Deep Kernel Learning (DKL) [2] autoencoder to reduce the dimensionality, followed by a DKL latent-state forward model that predicts the system dynamics.
Each sub-model produces an estimate of the amount of state variables the observed system is likely to have, and latent representations of both the state and the dynamics of the system, to bound the latent state space dimension and correct the system dynamics learnt on a reduced volume of high-fidelity data.
The major advantage of the proposed method lies in its substantial generality, both in terms of the nature of the complex systems it can learn, and the simplicity of the data sources it can adopt. The strategy is tested on a variety of cases, from the simple motion of a pendulum to a more challenging PDE problem. We believe that the achieved results in terms of efficiency, accuracy, and generality are extremely relevant also in view of data-driven physical modeling of more complex dynamical systems.

2. Preliminaries

In the following, we introduce some preliminary notions that will lay the foundations of our method, described in Section 3.

2.1. Gaussian Process Regression

To produce a data-driven surrogate model of a dynamical system and quantify uncertainty, we consider a nonparametric regression technique called Gaussian process regression (GPR). The

principle of GPR is that the prior knowledge of a given map $G(\mathbf{x})$ can be modelled by a GP $Z(\mathbf{x}) \sim \mathcal{GP}(m, k)$, with mean $m(\mathbf{x}) = E[Z(\mathbf{x})]$ and kernel function $k(\mathbf{x}, \mathbf{x}') = \sigma_k^2 r(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})$, where σ_k^2 and $\boldsymbol{\theta}$ are parameters to be estimated and the kernel class $r(\cdot)$ needs to be set.
Let $\mathcal{X} = (\mathbf{x}^{(1)}, ..., \mathbf{x}^{(N)}), \mathbf{x}^{(i)} \in \mathbb{R}^n$ be the training data inputs, and $\mathcal{Y} = (G(\mathbf{x}^{(1)}), ..., G(\mathbf{x}^{(N)}))$ the corresponding training targets. The predictive distribution of the GP evaluated at N^* test data points $\mathcal{X}^*$ is therefore

$$[Z(\mathcal{X}^*)|Z(\mathcal{X}) = \mathcal{Y}, \sigma^2, \boldsymbol{\theta}] \sim \mathcal{N}(\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$$

where

$$\boldsymbol{\mu}^* = \boldsymbol{\mu}_{\mathcal{X}^*} + K_{\mathcal{X}^*,\mathcal{X}}(K_{\mathcal{X},\mathcal{X}} + \sigma I)^{-1}\mathcal{Y}$$
$$\boldsymbol{\Sigma}^* = K_{\mathcal{X}^*,\mathcal{X}^*} + K_{\mathcal{X}^*,\mathcal{X}}(K_{\mathcal{X},\mathcal{X}} + \sigma I)^{-1}K_{\mathcal{X},\mathcal{X}^*}$$

2.2. Deep Kernel Learning

The GP hyperparameters σ and $\boldsymbol{\theta}$ are usually estimated by maximising the marginal likelihood, however this procedure can be very expensive when dealing with large, high-dimensional datasets. DKL [2] was developed to mitigate the limited scalability of traditional GPs, while maintaining their probabilistic features for UQ. More specifically, DKL embeds a nonlinear mapping $g(\mathbf{x}, \mathbf{w})$ from the input space to the feature space, given by an highly expressive deep architecture, into the kernel function:

$$k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}) \to k(g(\mathbf{x}, \mathbf{w}), g(\mathbf{x}', \mathbf{w}); \boldsymbol{\theta}, \mathbf{w})\,.$$

Since DKL still suffers from computational inefficiencies, Stochastic Variational Deep Kernel Learning [3] (SVDKL) was introduced to overcome these limitations and consider large datasets. SVDKL utilises variational inference to approximate the posterior distribution with the best fitting Gaussian to a set of inducing data points sampled from the posterior.

2.3. Multi-Fidelity methods

In our framework, we aim at considering a variety of data sources, which display different accuracy with respect to the true dynamical system, and exploit their correlation [4]. GPR with training data from different fidelity levels is known as co-kriging. The core idea is to leverage a large amount of low-fidelity (LF) data during training, in a setting in which a limited number of high-fidelity (HF) samples are available.

Assuming a linear correlation between the different fidelity levels, we can employ the Linear Model for Coregionalisation [5] (LMC), that expresses the prior of a hierarchy of L fidelities as

$$Z_l(\mathbf{x}) = \sum_{i=0}^{L-1} a_{l,i} u_i(\mathbf{x})\,, \quad l = 0, 1, ..., L-1,$$

namely, each level of solution $Z_l(\mathbf{x})$ can be written as a linear combination of L independent GPs $u_i \sim \mathcal{GP}(0, k_i(\cdot,\cdot))$. The vector $\mathbf{a}_i = (a_{0,i}, ..., a_{L-1,i})^\top$, $0 \le i \le L-1$, collects the weights of the corresponding GP component u_i and the matrix-valued kernel of the multi-fidelity GPR model assumes the form

$$K(\mathbf{x}, \mathbf{x}') = \sum_{i=0}^{L-1} \mathbf{a}_i \mathbf{a}_i^\top k_i(\mathbf{x}, \mathbf{x}')\,.$$

2.4. Levina-Bickel algorithm

Our framework involves dimensionality reduction and the latent state space dimension is an hyperparameter of the model. However, instead of fixing such hyperparameter to an arbitrary value, we prefer to automatically learn it by estimating the true system Intrinsic Dimensionality (ID) via the Levina-Bickel (LB) algorithm [6].
Consider n i.i.d. latent vectors $\mathbf{z}_1, ..., \mathbf{z}_n$ in $\mathbb{R}^p$, that represent an embedding of a lower-dimensional sample, i.e. $\mathbf{z}_i = g(\mathbf{s}_i)$, where $\mathbf{s}_i$ are sampled from an unknown smooth density f on $\mathbb{R}^{ID}$, with unknown $ID \le p$, and g is a continuous and sufficiently smooth mapping.
A key geometric observation is that the number of data points within distance R from any given $\mathbf{z}_i$ is proportional to R^{ID} when R is small [7]. On the basis of this observation, the LB algorithm derives the local ID estimator near $\mathbf{z}$ as $ID_{L-B}(\mathbf{z}) = \frac{1}{k-2}\sum_{j=1}^{k-1} \log \frac{T_k(\mathbf{z})}{T_j(\mathbf{z})}$, where $T_k(\mathbf{z})$ is the Euclidean distance between $\mathbf{z}$ and the its k-th nearest neighbor in $\mathbf{z}_1, ..., \mathbf{z}_n$. The global ID estimator is then calculated as

$$ID_{L-B} = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{k-2} \sum_{j=1}^{k-1} \log \frac{T_k(\mathbf{z}_i)}{T_j(\mathbf{z}_i)}\,.$$

3. Methods

Consider the nonlinear dynamical system:

$$\dot{\mathbf{s}}(t) = \mathcal{F}(\mathbf{s}(t)), \quad \mathbf{s}(t_0) = \mathbf{s}_0, \quad t \in [t_0, t_f],$$

where $\mathbf{s}(t) \in \mathcal{S} \subset \mathbb{R}^d$ is the state vector at time t, $\mathcal{F} : \mathcal{S} \to \mathcal{S}$ is a nonlinear function determining the system evolution given the current state $\mathbf{s}(t)$. Given L sets of N_l d_l-dimensional measurements $\mathbf{X}_l = (\mathbf{x}_l^1, \dots, \mathbf{x}_l^{N_l}) \in \mathcal{X}_l \subset \mathbb{R}^{d_l} \times \cdots \times \mathbb{R}^{d_l}$, with l being the fidelity level, $0 \leq l \leq L$, we consider the problem of identifying a meaningful representation of the hidden states and a surrogate model for $\mathcal{F}$, exploiting not only data from detailed HF simulations, but also partial or less accurate measurements, in a multi-fidelity fashion. In practice, we will build L surrogate (sub-)models Z_l, each one trained on a specific set of measurements $\mathbf{X}_l$ and leveraging the knowledge on the system acquired by the previous less accurate models Z_i, $i < l$.

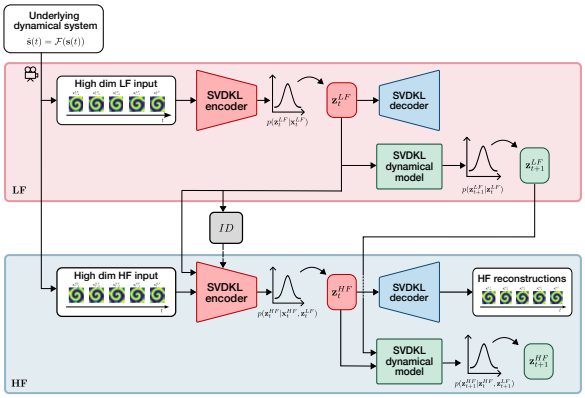


Figure 1: A scheme of the model.

3.1. Learning the hidden state

The first component of our model is a SVDKL encoder $E_l : \mathcal{X}_l \to \mathcal{L}_l$ that maps the high-dimensional measurements onto distributions over a low-dimensional latent space $\mathcal{L}_l$, where l indicates the fidelity level. A latent state sample can be obtained as:

$$z_{i,l}^{(t)} = Z_i^{E_l}(\mathbf{x}_l^{(t)}) + \varepsilon_{E_l}\,, \quad \varepsilon_{E_l} \sim \mathcal{N}(0, \sigma_{E_l}^2)\,,$$

where $z_{i,l}^{(t)}$ is the sample from the GP $Z_i^{E_l}$, with kernel $k(\cdot,\cdot;\boldsymbol{\theta}_{E_l,i})$ and mean μ, $g_{E_l}(\mathbf{x}_l^{(t)};\mathbf{w}^{E_l})$ is the feature vector output of the NN part of the SVDKL encoder E_l.
Since we have no access to the actual state values, the parameters $(\mathbf{w}_{E_l}, \boldsymbol{\theta}_{E_l}, \sigma_{E_l}^2)$ are optimised with a decoder $D_l : \mathcal{L}_l \times \mathcal{L}_{l-1} \to \mathcal{X}_l$ that reconstructs $\mathbf{x}_l$ given $\tilde{\mathbf{z}}_l = \mathbf{z}_l + \rho \mathbf{z}_{l-1}$. While the SVDKL encoder E_l learns $p(\mathbf{z}_l^{(t)}|\mathbf{x}_l^{(t)})$, the decoder D_l learns the inverse mapping $p(\hat{\mathbf{x}}_l^{(t)}|\tilde{\mathbf{z}}_l^{(t)})$, where $\hat{\mathbf{x}}_l^{(t)}$ is the reconstruction of $\mathbf{x}_l^{(t)}$. In particular, we minimise the following loss function:

$$\text{loss}_{E_l}(\mathbf{w}_{E_l}, \boldsymbol{\theta}_{E_l}, \sigma_{E_l}^2, \boldsymbol{\theta}_{D_l}) = \mathbb{E}_{\mathbf{x}_l^{(t)} \sim \mathbf{X}_l}[-\log p(\hat{\mathbf{x}}_l^{(t)}|\tilde{\mathbf{z}}_l^{(t)})]\,,$$

where $\hat{\mathbf{x}}_l^{(t)}|\tilde{\mathbf{z}}_l^{(t)} \sim \mathcal{N}(\mu_{\hat{\mathbf{x}}_l^{(t)}}, \Sigma_{\hat{\mathbf{x}}_l^{(t)}})$.

3.2. Learning the latent dynamics

The second component of each sub-model is a surrogate model $F_l : \mathcal{L}_l \times \mathcal{L}_{l-1} \to \mathcal{L}_l$, still based on a SVDKL architecture, that aims to learn and predict the system evolution given the latent state variables sampled from $\mathcal{L}_l$ and the next latent state $\mathbf{z}_{l-1}^{(t+1)}$ sampled from $\mathcal{L}_{l-1}$. The next latent states, at fidelity level l, $\mathbf{z}_l^{(t+1)}$ can be sampled with F_l:

$$z_{i,l}^{(t+1)} = Z_i^{F_l}(\mathbf{z}_l^{(t)}, \mathbf{z}_{l-1}^{(t+1)}) + \varepsilon_{F_l}\,,\ \varepsilon_{F_l} \sim \mathcal{N}(0, \sigma_{F_l}^2)\,,$$

where $z_{i,l}^{(t+1)}$ is sampled from the GP $Z_i^{F_l}$, and $g_{F_l}(\mathbf{z}_l^{(t)}, \mathbf{z}_{l-1}^{(t+1)}; \mathbf{w}^{F_l})$ is the feature vector output of the NN part of the SVDKL dynamical model. Due to the unsupervised nature of our strategy, the learning process of the dynamics employs the encoding of the measurement $\mathbf{x}_l^{(t+1)}$ through the SVDKL encoder E_l to the distribution $p(\mathbf{z}_l^{(t+1)}|\mathbf{x}_l^{(t+1)})$, and uses such a distribution as target for $p(\mathbf{z}_l^{(t+1)}|\mathbf{z}_l^{(t)}, \mathbf{z}_{l-1}^{(t+1)})$ produced by the dynamical model F_l. In particular, F_l is trained by minimising the Kullback-Leibler divergence between such distributions:

$$\text{loss}_{F_l}(\mathbf{w}_{F_l}, \boldsymbol{\theta}_{F_l}, \sigma_{F_l}^2) = \mathbb{E}_{\mathbf{x}_l^{(t)}, \mathbf{x}_l^{(t+1)} \sim \mathbf{X}_l}[\mathrm{KL}[p(\mathbf{z}_l^{(t+1)}|\mathbf{x}_l^{(t+1)}) || \ p(\mathbf{z}_l^{(t+1)}|\mathbf{z}_l^{(t)}, \mathbf{z}_{l-1}^{(t+1)})]]\,,$$

Moreover, we reintroduce the same loss term used to optimise the SVDKL AE parameters, to minimise the error between the reconstruction $\hat{\mathbf{x}}_l^{(t+1)}$ produced by the dynamical model, and the one form the SVDKL AE applied to $\mathbf{x}_l^{(t+1)}$.

3.3. Training of the models

The two components E_l and F_l are jointly trained, allowing the gradients of the dynamical model F_l to flow through the encoder E_l as well.

The loss function is given by:

$$\begin{aligned}\text{loss}_l(\mathbf{w}_{E_l}, \boldsymbol{\theta}_{E_l}, \sigma^2_{E_l}, \boldsymbol{\theta}_{D_l}, \mathbf{w}_{F_l}, \boldsymbol{\theta}_{F_l}, \sigma^2_{F_l}) = \\ \mathbb{E}_{\mathbf{x}_l^{(t)}, \mathbf{x}_l^{(t+1)} \sim \mathbf{X}_l}[-\log p(\hat{\mathbf{x}}_l^{(t)} | \mathbf{z}_l^{(t)}, \mathbf{z}_{l-1}^{(t)}) \\ + \beta \text{KL}[p(\mathbf{z}_l^{(t+1)} | \mathbf{x}_l^{(t+1)}) || p(\mathbf{z}_l^{(t+1)} | \mathbf{z}_l^{(t)}, \mathbf{z}_{l-1}^{(t+1)}] \\ - \log p(\hat{\mathbf{x}}_l^{(t+1)} | \mathbf{z}_l^{(t+1)}, \mathbf{z}_{l-1}^{(t+1)})] \,.\end{aligned}$$

Since the two SVDKL components exploit variational inference to approximate the aforementioned posterior distributions, we add two extra terms to the loss function, one for each SVDKL component, of the form

$$\text{loss}_{var}(\mathbf{w}, \boldsymbol{\theta}) = \text{KL}[p(\mathbf{v}) || q(\mathbf{v})] \,, \qquad (1)$$

where $p(\mathbf{v})$ is the posterior to be approximated over the inducing points $\mathbf{v}$, and $q(\mathbf{v})$ represents an approximating candidate distribution.
Once the sub-model at fidelity level l is trained, it is evaluated on a portion of the training data $\mathbf{X}_l$ available also at level of fidelity $l+1$. The learnt latent representations of both the hidden state and the dynamics are used as additional inputs during the training of the sub-model $l+1$.

3.4. Intrinsic dimension of the system

In our framework, the Levina-Bickel algorithm is applied to the latent representations of the states $\mathbf{z}_l^t$ and $\mathbf{z}_l^{t+1}$ obtained from the SVDKL AE and the latent forward dynamics learnt by the SVDKL dynamical model. In particular, when constructing the sub-model at level $l+1$, the ID estimate at level l is employed as dimension of the latent space $|\mathbf{z}|$, to restrict the latent space of the new sub-model.

4. Numerical Results

The model performance is assessed on measurements generated by either low-dimensional dynamical systems, or high-dimensional dynamical systems from PDE discretisation.
In the following, we present the main results related to the diffusion-advection problem given by

$$\frac{\partial \omega}{\partial t} + \mu \left(\frac{\partial \psi}{\partial x} \frac{\partial \omega}{\partial y} - \frac{\partial \psi}{\partial y} \frac{\partial \omega}{\partial x} \right) = d \nabla^2 \omega,$$
$$\nabla^2 \psi = \omega \,,$$

defined over a spatial domain $(x, y) \in [-L, L]^2$ and a time span $t \in [0, T]$. Here $\omega(x, y, t)$ and $\psi(x, y, t)$ represent the vorticity and stream function, respectively, and d is the diffusion coefficient. For $(x, y) \in [-L, L]^2$, we consider the following initial condition of vorticity:

$$\omega(x, y, 0) = \exp\left(-2x^2 - \frac{y^2}{20}\right) .$$

Our goal is to approximate the time-dependent vorticity field ω as the parameter μ varies over $\mathcal{P} = [1, 5]$. We adopt two fidelity levels that differ in the spacial resolution of discretization and in length of the time window, with $n^{LF} < n^{HF}$ and $T^{LF}_{train} > T^{HF}_{train}$.ì – see Figure 2. The system is measured for an additional time window $[T^{HF}_{train}, T^{HF}_{train} + T_{test}]$.

(a) Low fidelity sample (b) High fidelity sample

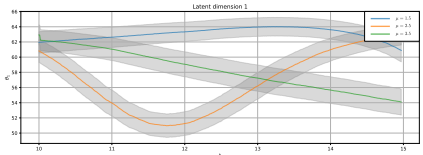
Figure 2: Samples from the diffusion-advection dataset.

4.1. ID and latent variables

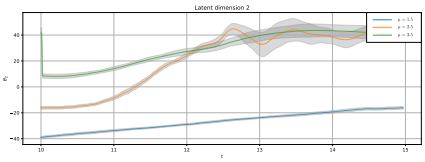
The Levina-Bickel algorithm applied on the LF latent representation estimates $ID = 2$, which is compatible with our theoretical knowledge of problem. Figure 3 shows the latent variables $\theta_1^{HF}(t)$ and $\theta_2^{HF}(t)$ as functions of time, for each value of μ, with uncertainty bands of $\pm$ two standard deviation in the predictive distribution.

4.2. Reconstruction and forward prediction

We illustrate now the effectiveness of the model in reconstructing the frames and predicting forward in time, for the testing parameter value $\mu = 1.5$, $t \in [10, 15]$. Additionally, we fix $dt = 0.01$, $d = 0.001$, $L = 5$, $n^{LF} = 32$, $n^{HF} = 100$.
Figure 4 shows the reconstruction of the frame $\mathbf{x}_t^{HF}$, produced by the SVDKL Autoencoder, and the one-step forward prediction of $\mathbf{x}_{t+dt}^{HF}$, generated by the DKL Dynamical model, with their respective absolute errors. From a qualitative point of view, both the reconstructions and the predictions are consistently accurate across



(a) $\theta_1^{HF}(t)$



(b) $\theta_2^{HF}(t)$

Figure 3: Latent variables of the HF autoencoder during the test time window, for each value of the parameter μ.

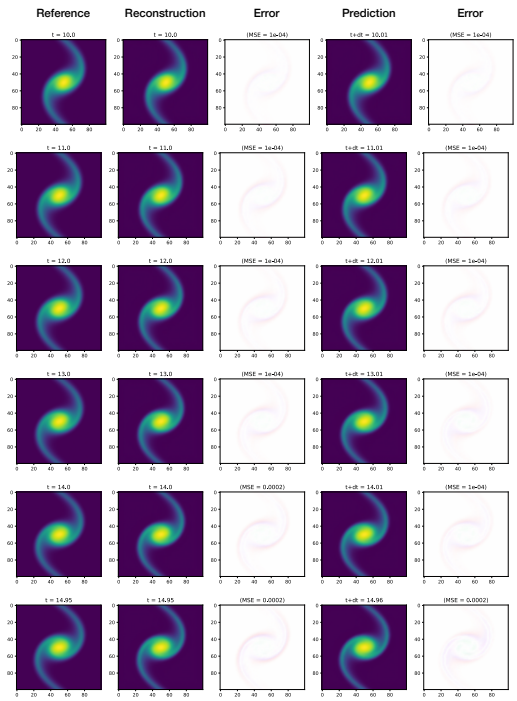


Figure 4: Reconstructions and predictions one-step forward in time, with respective absolute errors, for $\mu = 1.5$.

the entire testing time window. Indeed, the MSE is consistently small, $< 2 \cdot 10^{-4}$, as shown in Figure 5, quantitatively corroborating the first visual impression.

4.3. Extrapolation in time

If we iterate the one-step forward prediction, by feeding the predicted latent representation to the next iteration, we can extrapolate forward in

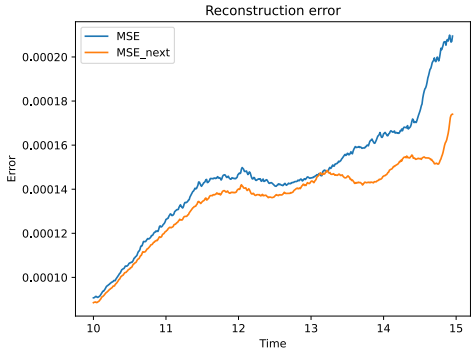


Figure 5: MSEs of the reconstruction and forward prediction for $\mu = 1.5$, with respect to the relative true measurements, as function of time.

time. Figure 6 shows some extrapolated frames, starting from the observation at time $t = 10$. Qualitatively speaking, the prediction remains consistent for the entire time window of 2 seconds (200 iterations), while gradually decreasing its accuracy. Figure 7 shows the MSE error of the extrapolation with respect to the actual measurements: it increases exponentially for the first few iterations, and then linearly until the end, but still staying below the $5 \cdot 10^{-3}$ threshold.

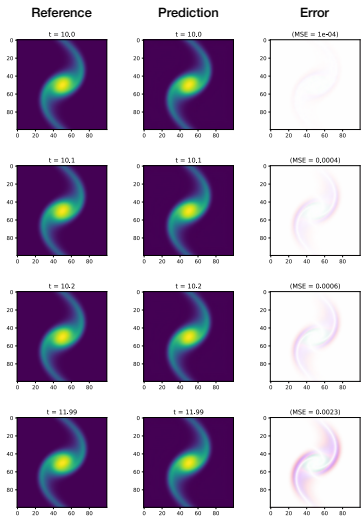


Figure 6: Extrapolation in time for $\mu = 1.5$, for 200 iterations, with the respective absolute error.

4.4. Some considerations

Overall, the results exhibit notable consistence and accuracy both when reconstructing the frame and predicting the dynamics one-step for-

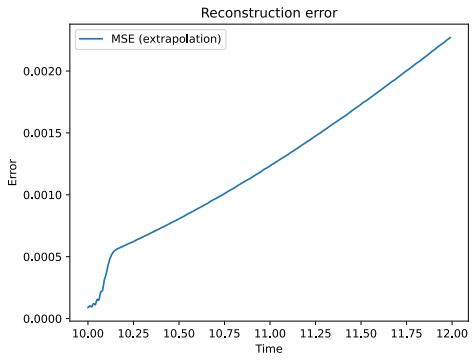


Figure 7: MSE of the extrapolation in time, for $\mu = 1.5$.

ward in time, highlighting the efficiency of both the comprehensive architecture and the Levina-Bickel estimator. The efficiency of the model is also highlighted by the significantly low uncertainty exhibited when predicting the latent state.
At the cost of some foreseeable and minor defects where the gradient is larger, our framework still performs reasonably well when extrapolating over the parameter space. In general, the MSEs is bounded within acceptable values. Additionally, the more challenging task of extrapolation in time is also tackled with distinctive results.

5. Conclusions

With this work, we propose a comprehensive framework for unsupervised data-driven discovery of low dimensional dynamical systems. By conjugating the expressive capability and flexibility of SVDKL with MF schemes, the model is able to exploit large datasets of high dimensional measurements coming from a variety of data sources, exhibiting different degrees of fidelity to the true system.
The model proves to be effective at dimensionality reduction, uncertainty quantification and latent dynamics learning, showing good capabilities of generating interpretable latent representations of a large class of problems. In particular, the introduction of the Levina-Bickel algorithm enables the estimation of an efficient and compact latent state representation, close to the true state space.
Additionally, our framework allows accurate data-driven unsupervised learning from ordinary RGB videos, to the best of our knowledge for the first time in a MF context, effectively replacing a considerable amount of high-fidelity data with the less expensive counterpart, while maintaining notable accuracy and reducing the overall cost of measurements.

References

[1] Nicolò Botteghi, Mengwu Guo, and Christoph Brune. Deep Kernel Learning of Dynamical Models from High-Dimensional Noisy Data. *Scientific Reports*, 12(1):21530, December 2022. arXiv:2208.12975 [cs, stat].

[2] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P. Xing. Deep Kernel Learning. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 370–378. PMLR, May 2016.

[3] Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P. Xing. Stochastic Variational Deep Kernel Learning, November 2016. arXiv:1611.00336 [cs, stat].

[4] Mengwu Guo, Andrea Manzoni, Maurice Amendt, Paolo Conti, and Jan S. Hesthaven. Multi-fidelity regression using artificial neural networks: efficient approximation of parameter-dependent output quantities. *Computer Methods in Applied Mechanics and Engineering*, 389:114378, February 2022. arXiv:2102.13403 [cs, math].

[5] Mauricio A. Alvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for Vector-Valued Functions: a Review, April 2012. arXiv:1106.6251 [cs, math, stat].

[6] Elizaveta Levina and Peter Bickel. Maximum Likelihood Estimation of Intrinsic Dimension. In *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004.

[7] Boyuan Chen, Kuang Huang, Sunand Raghupathi, Ishaan Chandratreya, Qiang Du, and Hod Lipson. Discovering State Variables Hidden in Experimental Data, December 2021. arXiv:2112.10755 [physics].