Extracting *Interpretable Physical Parameters* from Spatiotemporal Systems using Unsupervised Learning

Peter Y. Lu, Samuel Kim, Marin Soljačić

March Meeting 2020

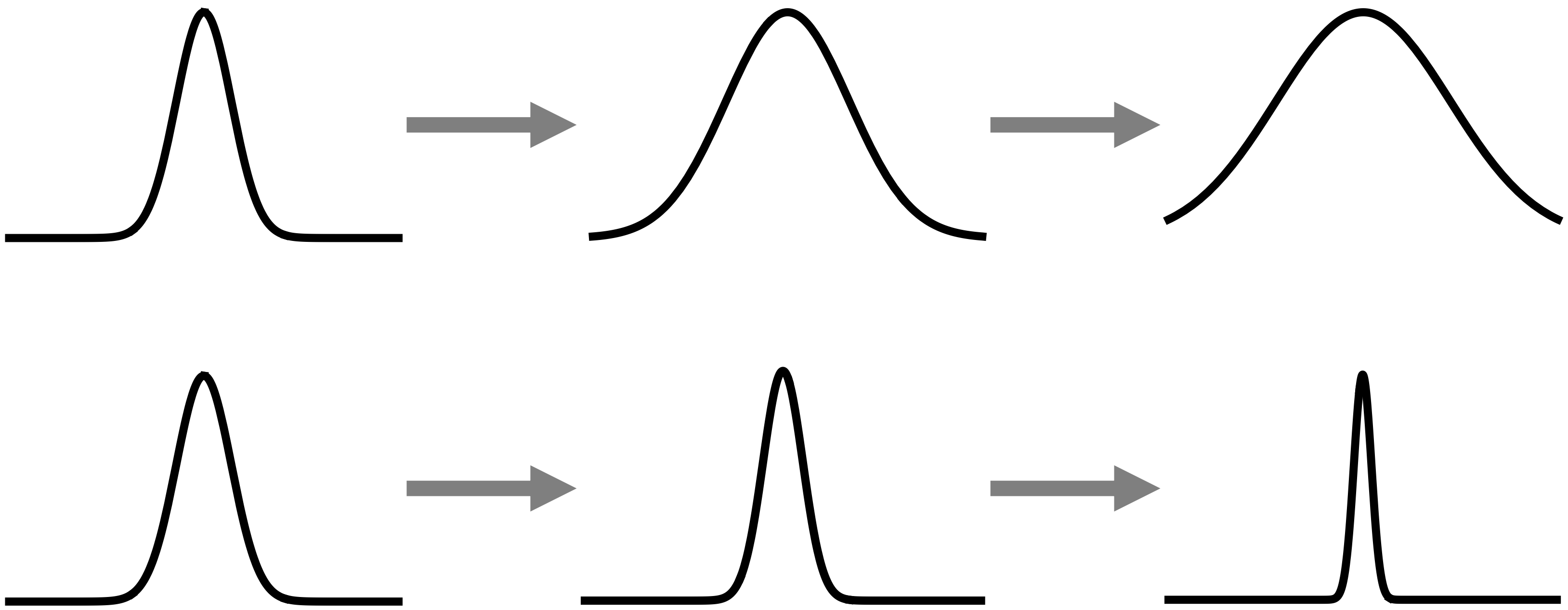
arXiv:1907.06011

Massachusetts Institute of Technology

Physics

Uncontrolled Variables in Dynamics Data

Consider a dataset where uncontrolled variables cause each example to have differing dynamics.

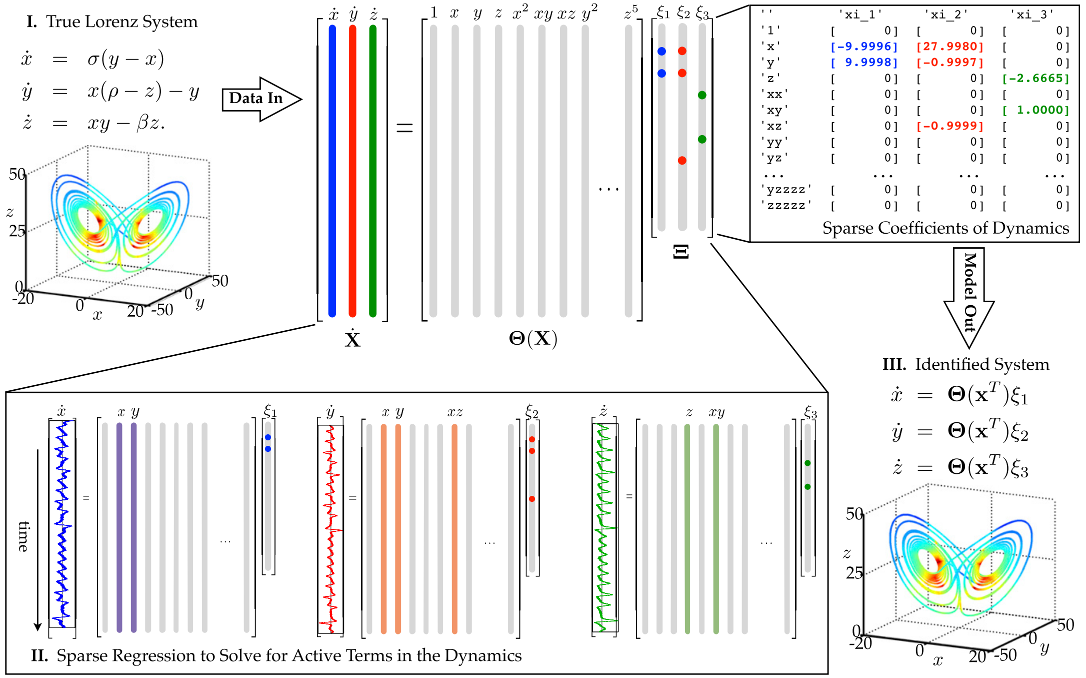


arXiv:1907.06011

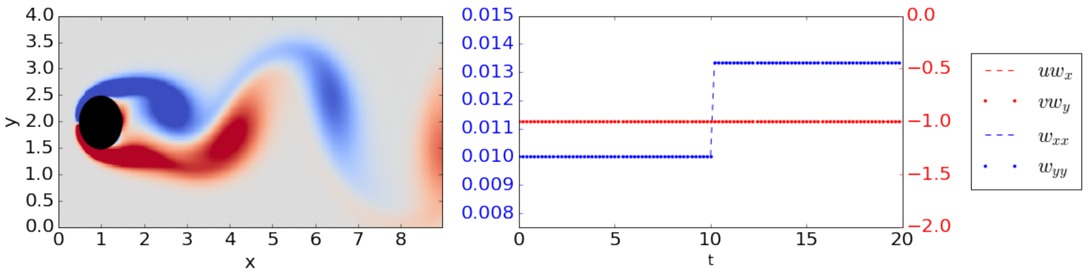
Previous Work: Identifying Parameters

- Previous approaches include sparse identification methods (SINDy).
- Related works include unsupervised learning of object properties.

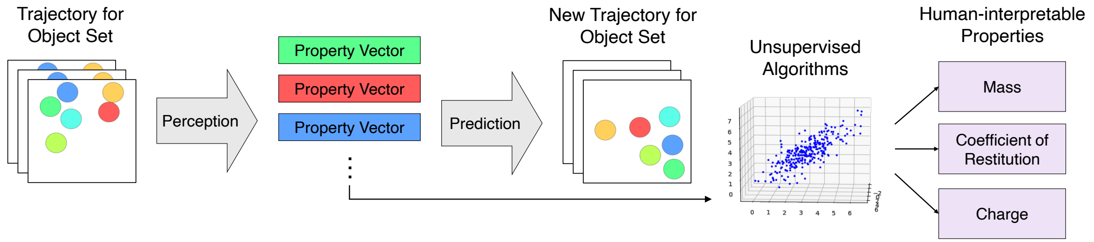
Steven L. Brunton, Joshua L. Proctor, J. Nathan Kutz (2016).
PNAS, 113(15), 3932-3937.



Samuel Rudy, Alessandro Alla, Steven L. Brunton, J. Nathan Kutz (2019).
SIAM J. Appl. Dyn. Syst., 18(2), 643–660.



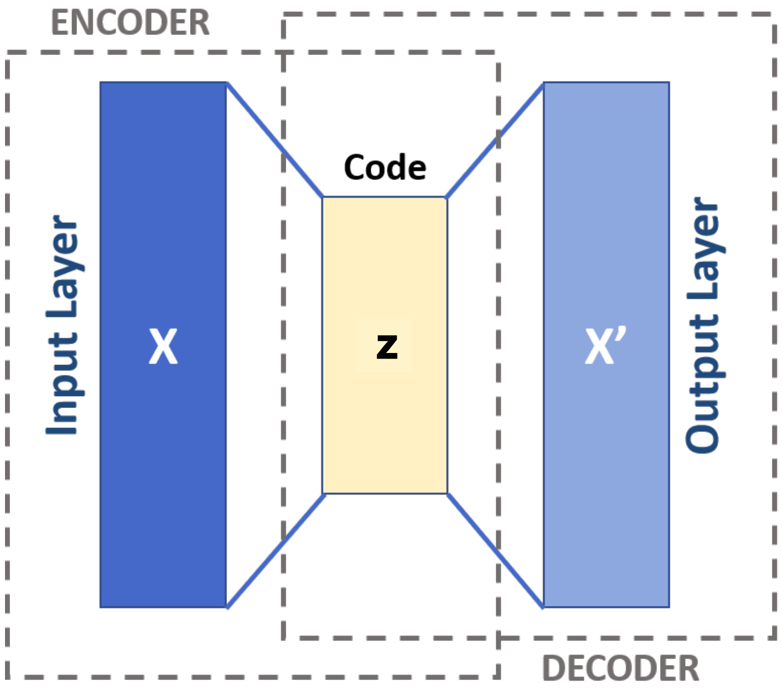
David Zheng, Vinson Luo, Jiajun Wu, Joshua B. Tenenbaum (2018).
UAI 2018 Proceedings, arXiv:1807.09244.



Unsupervised Learning & Interpretability

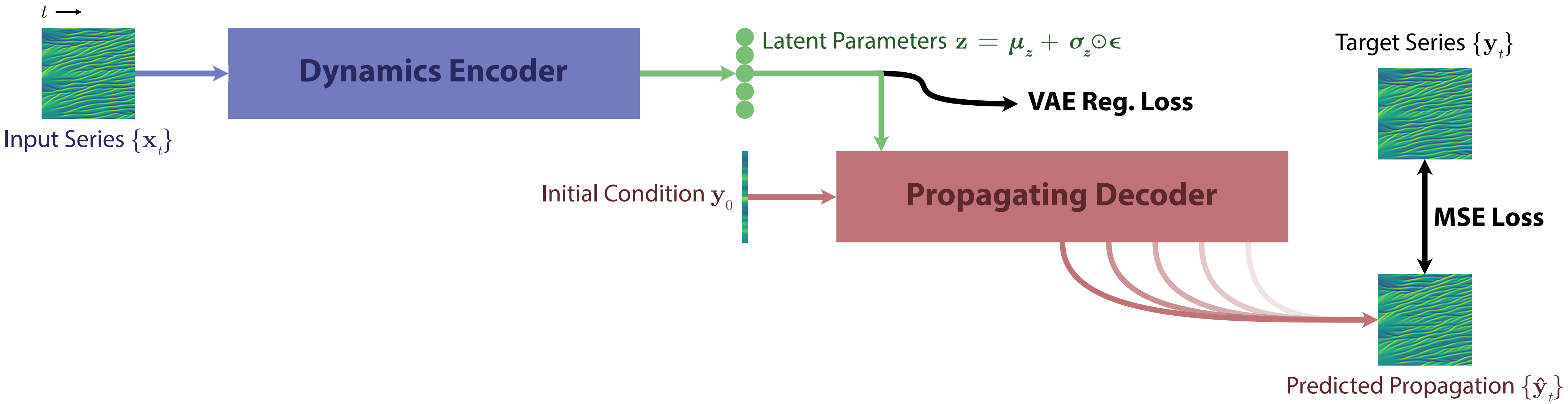
Goal: Given data with varying dynamics due to uncontrolled variables, extract *interpretable parameters* that characterize the observed dynamics.

- Assume no explicit equation or model.
- Use variational autoencoder (VAE) to produce interpretable latent representations.
- Physics-informed architectures provide inductive bias.



arXiv:1907.06011

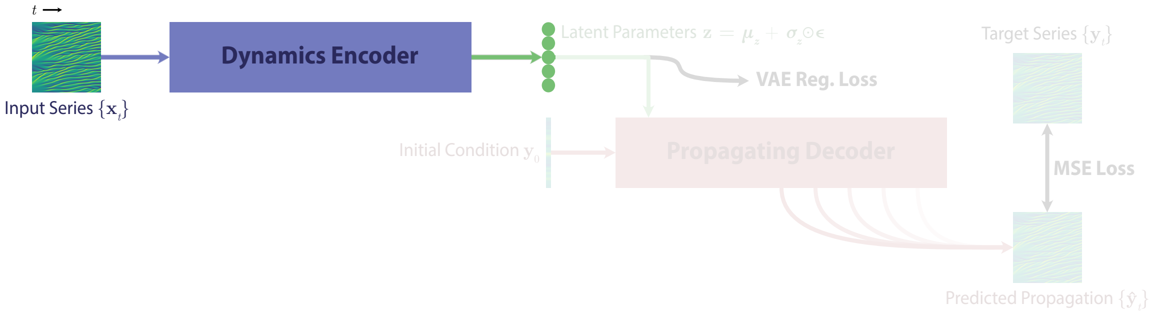
Architecture Overview

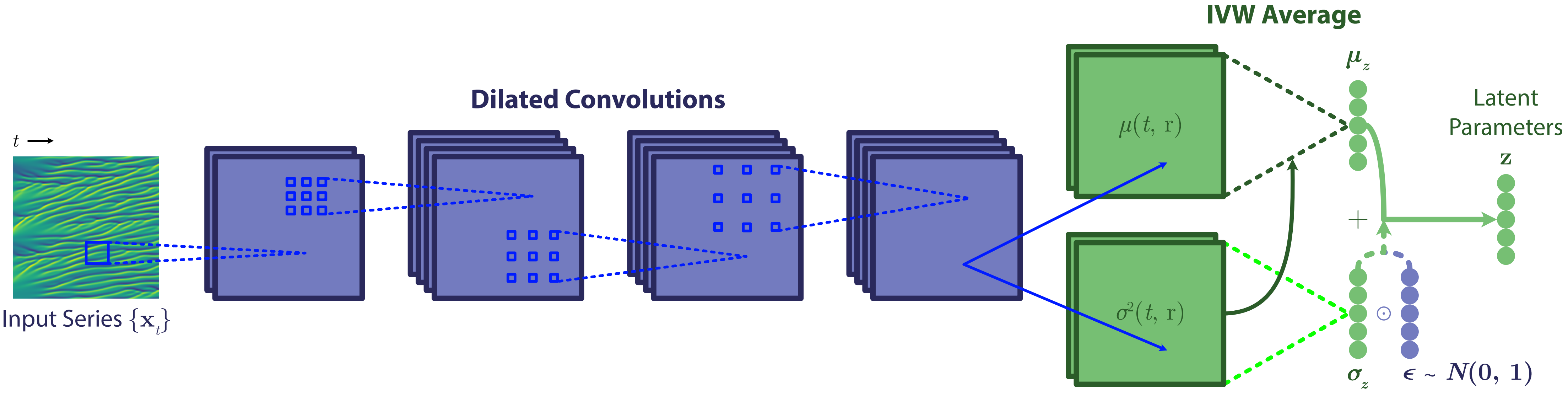


- Based on variational autoencoder (VAE): independent latent parameters.
- **Encoder** extracts latent physical parameters.
- **Decoder** propagates the system forward in time (simulator).

arXiv:1907.06011

Encoder Architecture

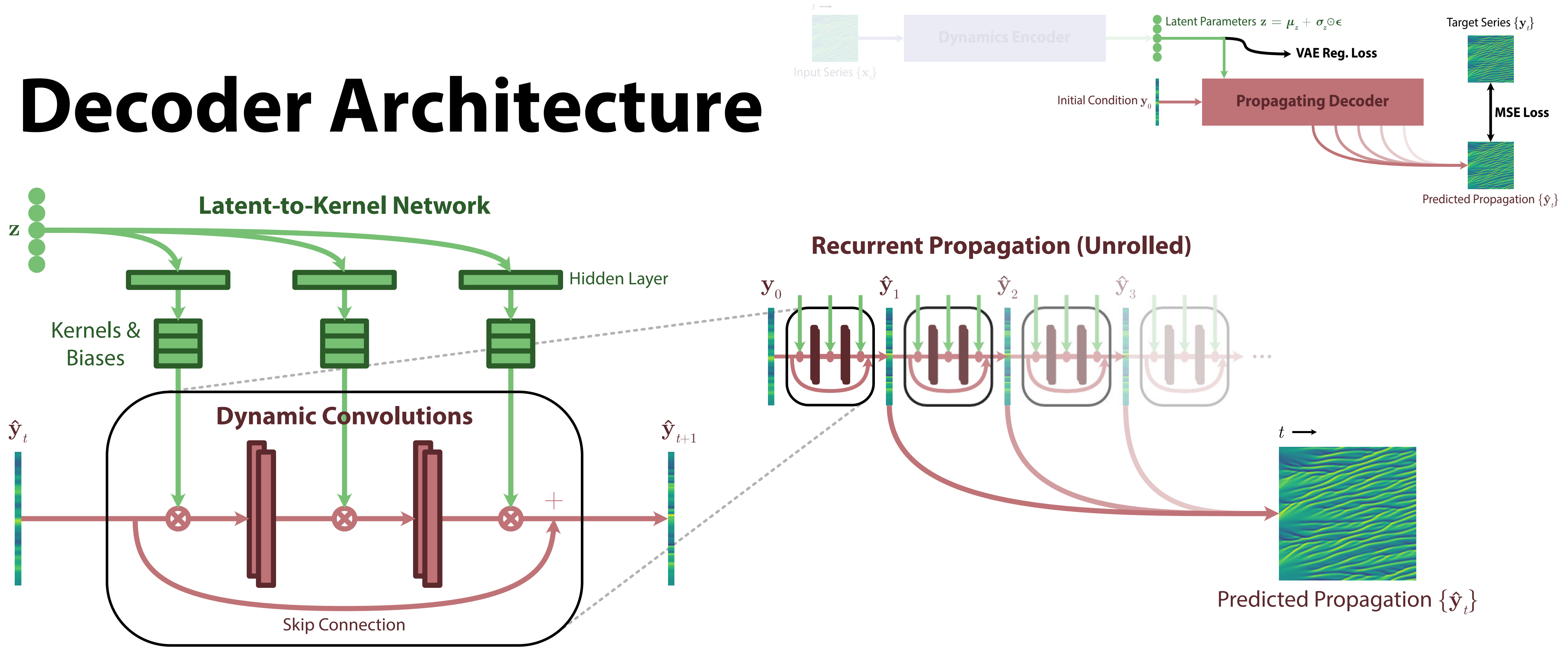




- Common architecture in computer vision tasks.
- Averaging to generalize to inputs of different sizes.
- Latent parameters **z** sampled from learned distribution $N(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z^2)$.

arXiv:1907.06011

Decoder Architecture



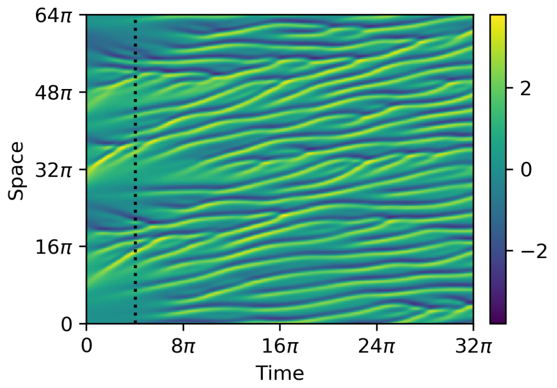
- Decoder network propagates state **ŷ** from time **t** → **t+1**.
- Latent parameters **z** directly parameterize propagation dynamics.

arXiv:1907.06011

Simulated Datasets + Noise

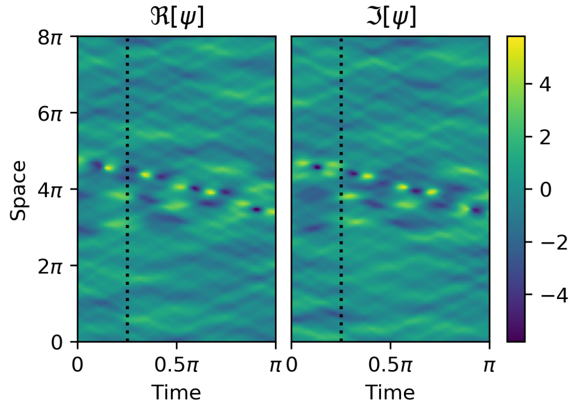
1D Kuramoto–Sivashinsky (KS)

$$\frac{\partial u}{\partial t} = -\gamma\,\partial_x^4 u - \partial_x^2 u - u\,\partial_x u$$



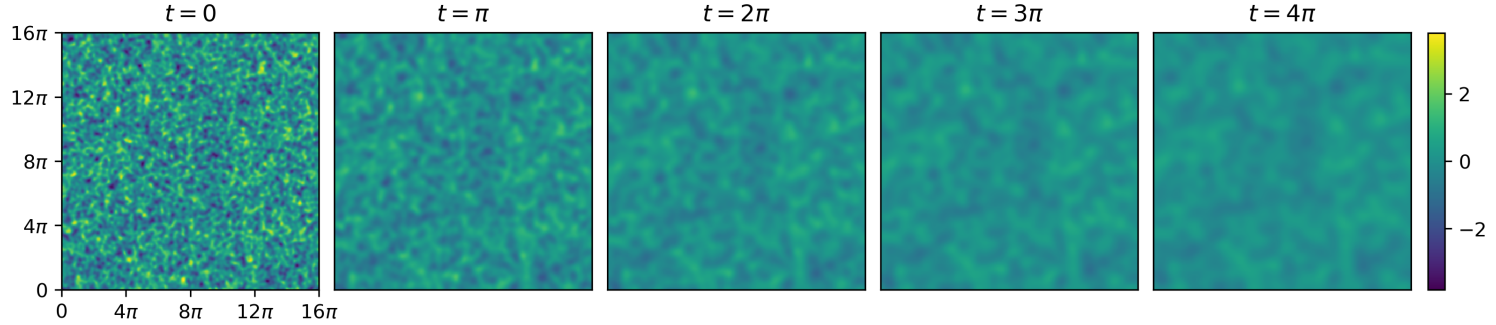
1D Nonlinear Schrödinger (NS)

$$i\partial_t \psi = -\frac{1}{2}\partial_x^2 \psi + \kappa|\psi|^2\psi$$



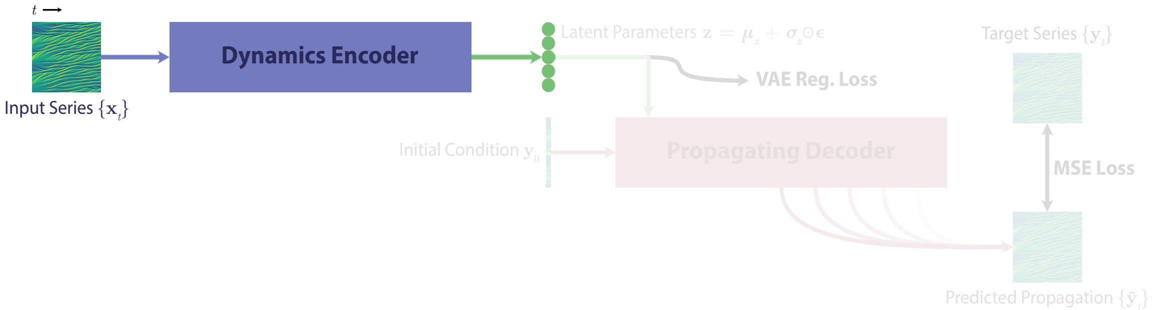
2D Convection–Diffusion (CD)

$$\frac{\partial c}{\partial t} = D\nabla^2 c - \vec{v}\cdot\nabla c$$

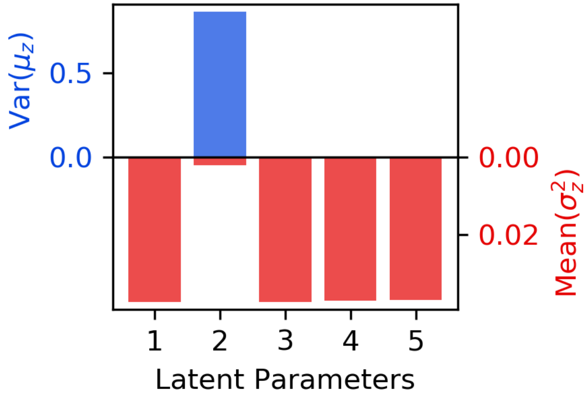


arXiv:1907.06011

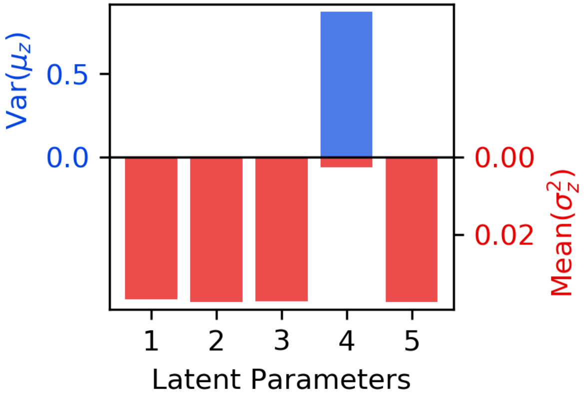
Identifying Relevant Parameters



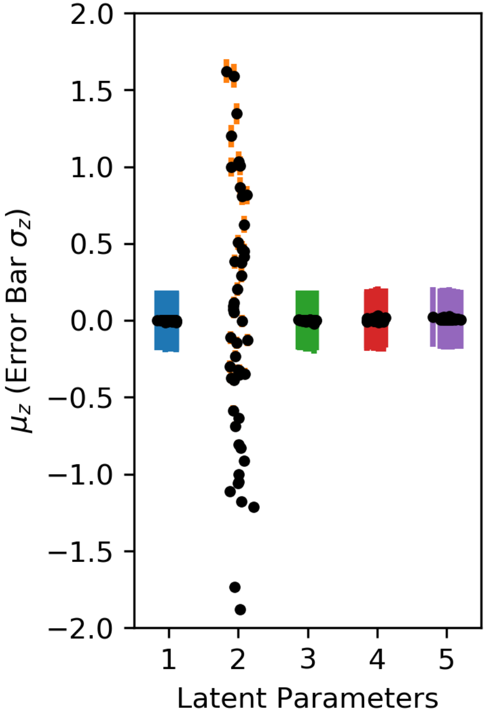
After training, examine encoded parameter distributions $\mathbf{z} \sim N(\boldsymbol{\mu}_z, \boldsymbol{\sigma}_z^2)$ to identify relevant parameters.



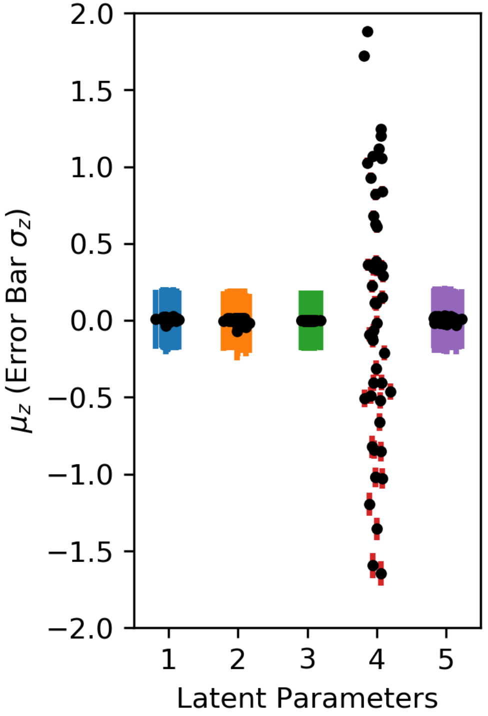
(a) KS, No Noise



(b) KS, $\sigma = 0.1$ Noise



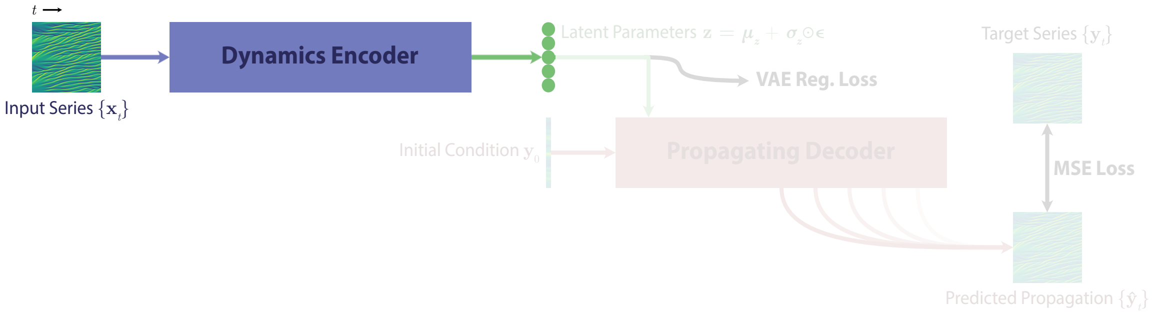
(a) KS, No Noise



(b) KS, $\sigma = 0.1$ Noise

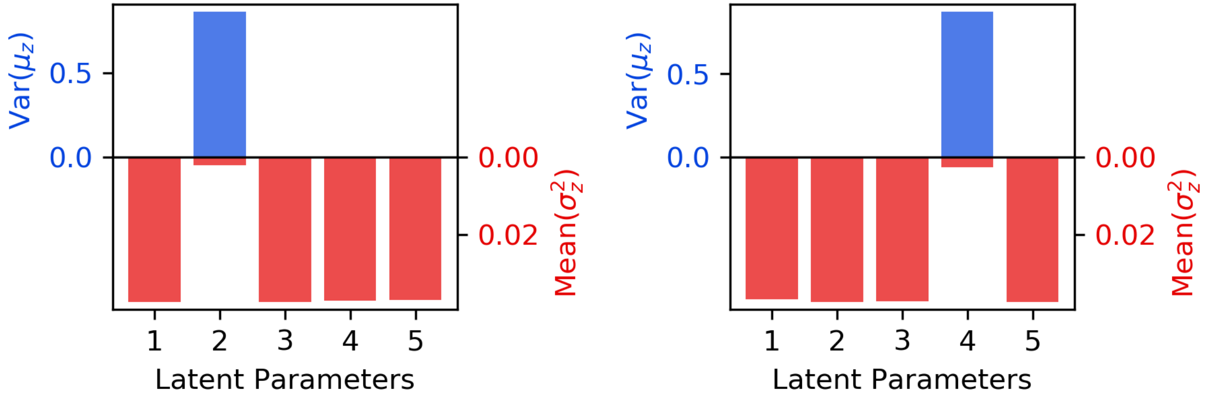
arXiv:1907.06011

Encoder Parameter Extraction

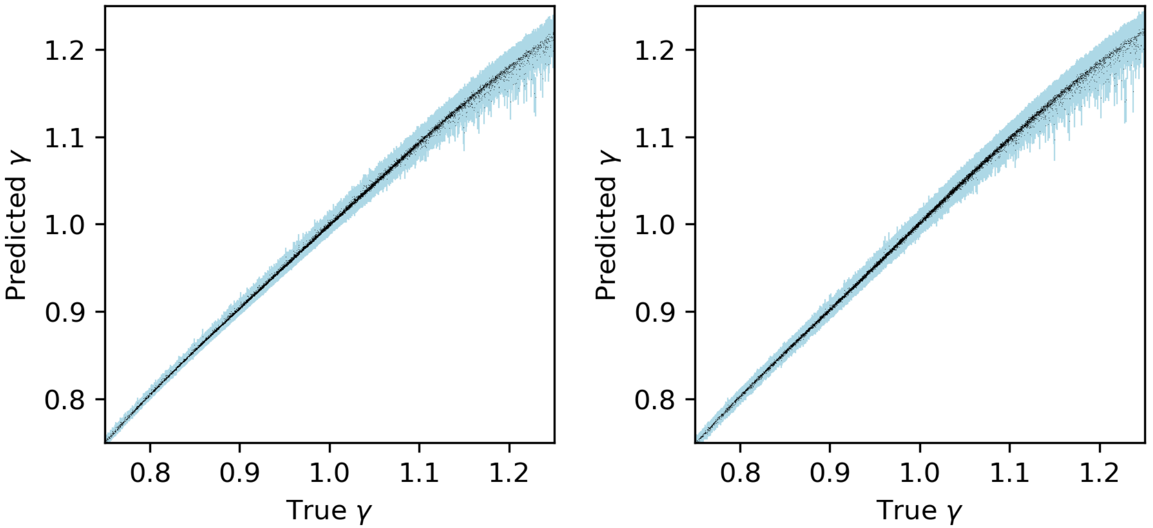


1D Kuramoto–Sivashinsky (KS)

$$\frac{\partial u}{\partial t} = -\gamma\, \partial_x^4 u - \partial_x^2 u - u\, \partial_x u$$



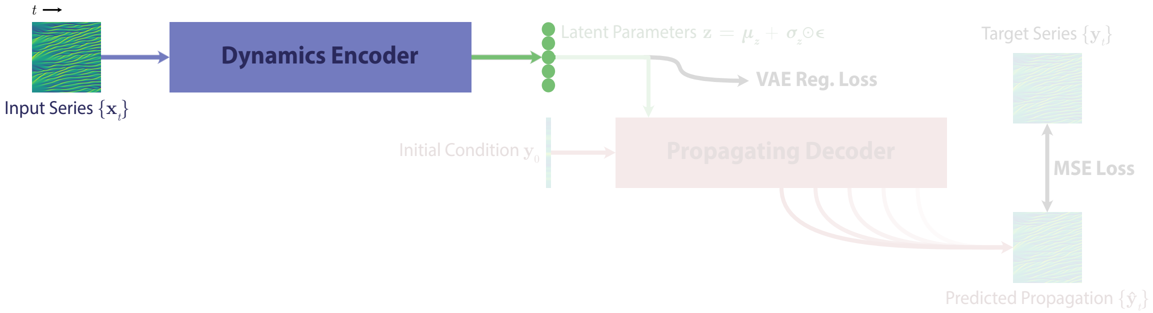
(a) KS, No Noise **(b) KS, $\sigma = 0.1$ Noise**



(a) KS, No Noise **(b) KS, $\sigma = 0.1$ Noise**

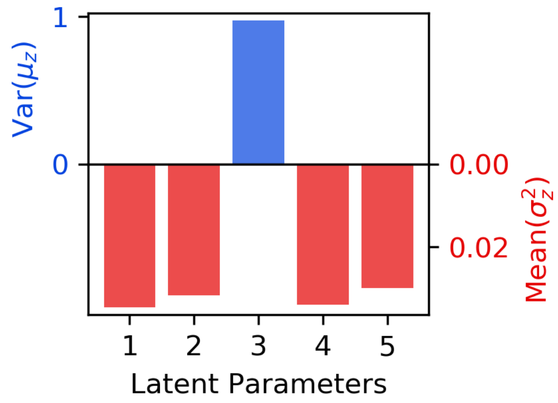
arXiv:1907.06011

Encoder Parameter Extraction

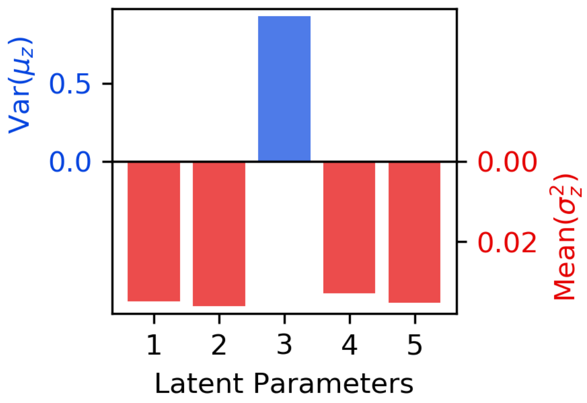


1D Nonlinear Schrödinger (NS)

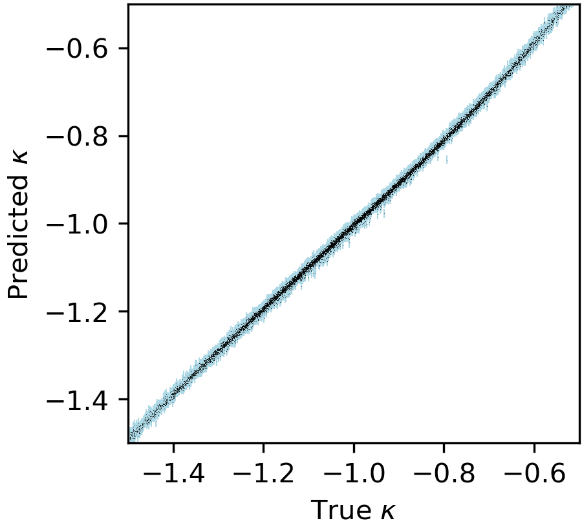
$$i\partial_t \psi = -\frac{1}{2}\partial_x^2 \psi + \kappa|\psi|^2 \psi$$



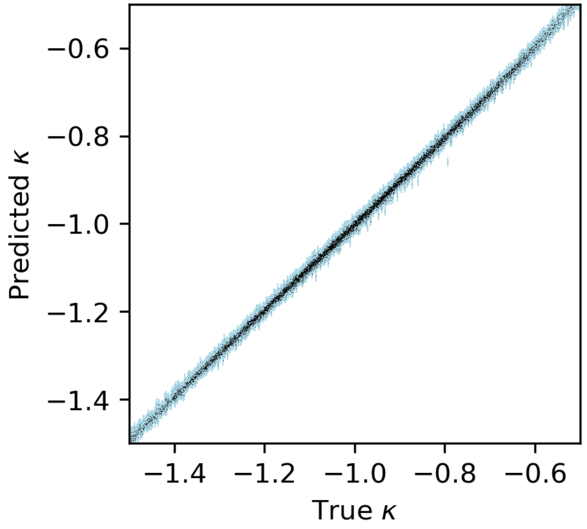
(c) NS, No Noise



(d) NS, $\sigma = 0.1$ Noise



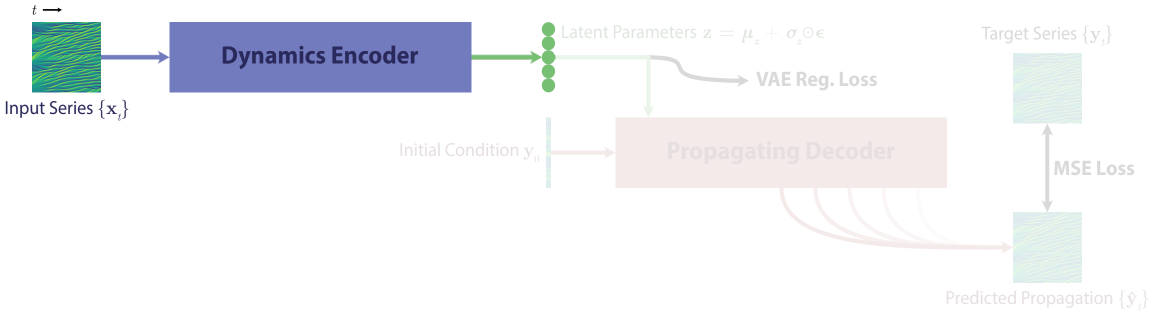
(c) NS, No Noise



(d) NS, $\sigma = 0.1$ Noise

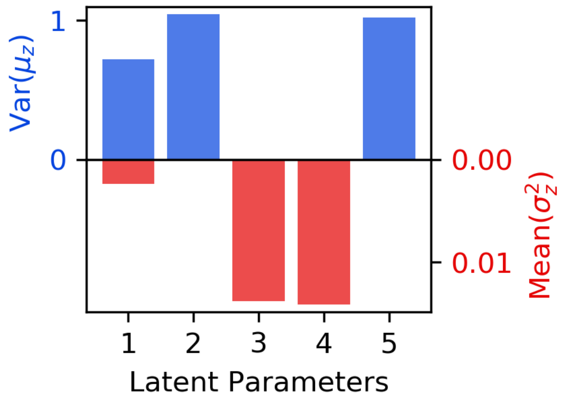
arXiv:1907.06011

Encoder
Parameter Extraction

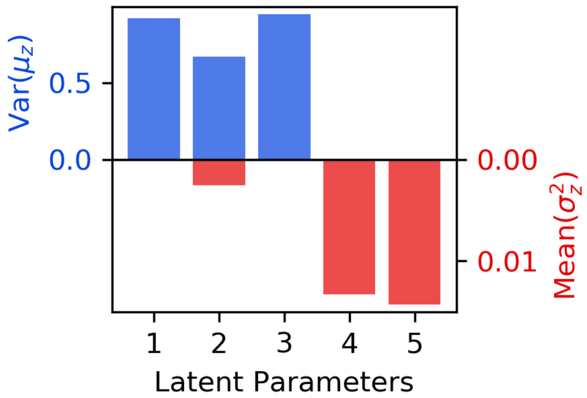


2D Convection–Diffusion (CD)

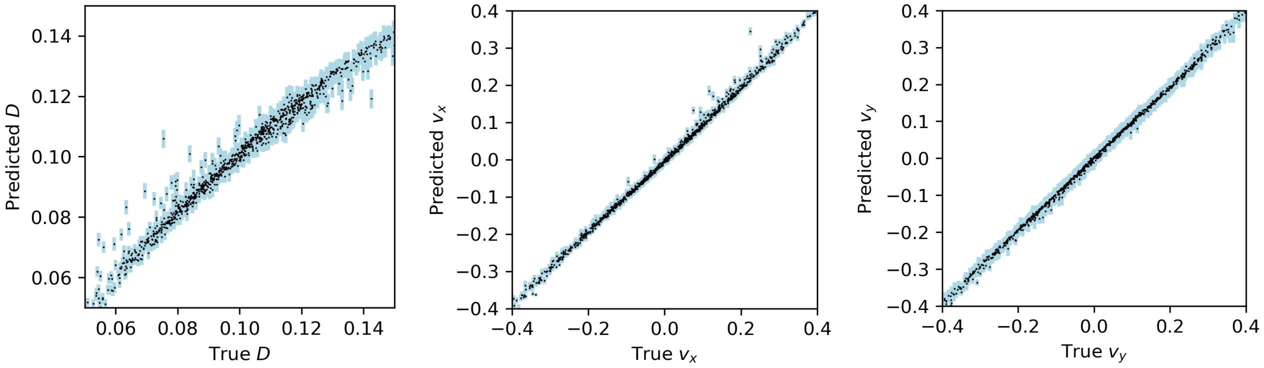
$$\frac{\partial c}{\partial t} = D\nabla^2 c - \vec{v}\cdot\nabla c$$



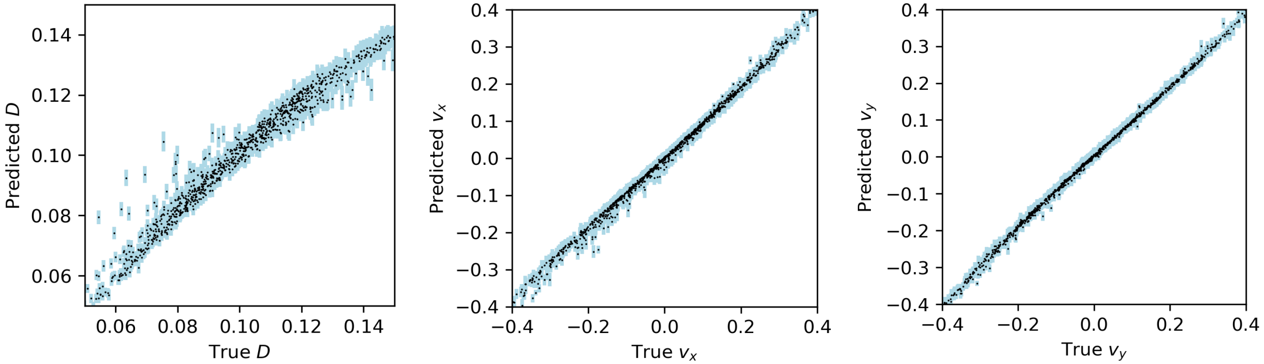
(e) CD, No Noise



(f) CD, $\sigma = 0.1$ Noise



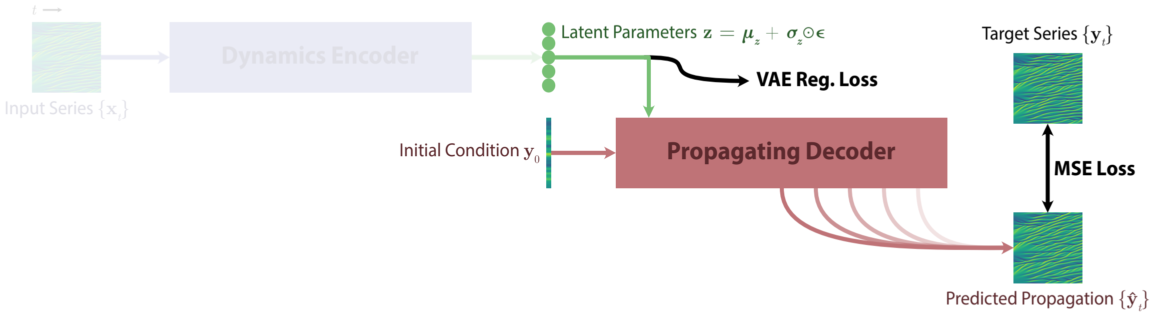
(e) CD, No Noise



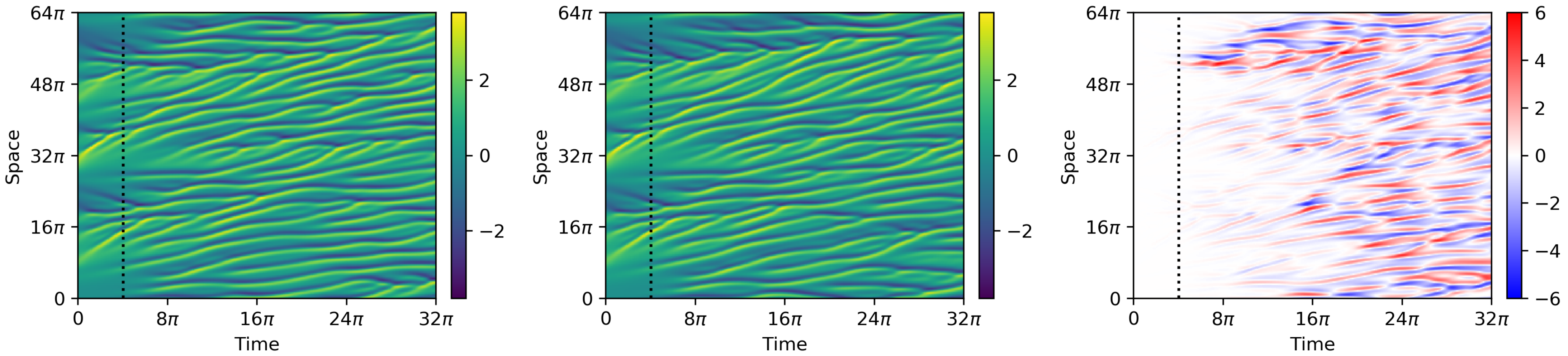
(f) CD, $\sigma = 0.1$ Noise

arXiv:1907.06011

Decoder
Tunable Model



Reasonable prediction performance and good generalization.

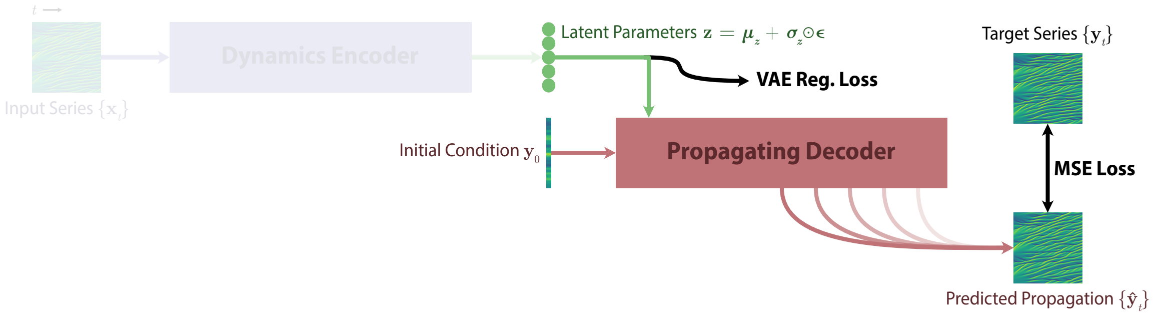


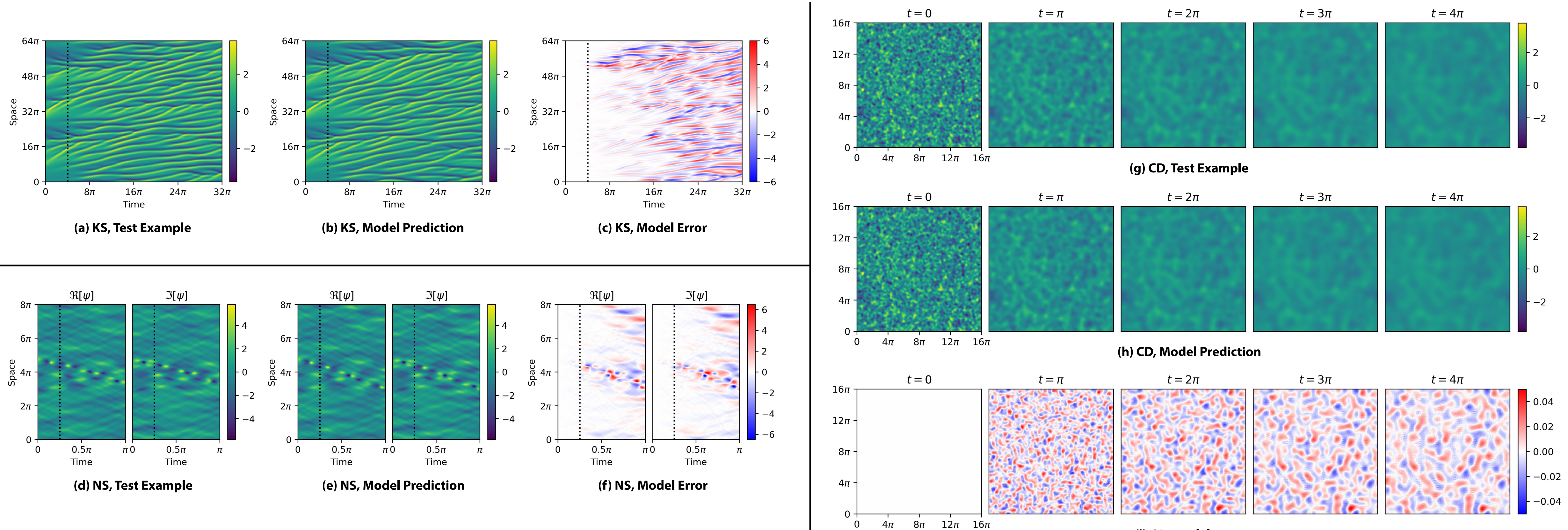
(a) KS, Test Example (b) KS, Model Prediction (c) KS, Model Error

arXiv:1907.06011

Decoder Prediction Examples

Example predictions from each dataset.

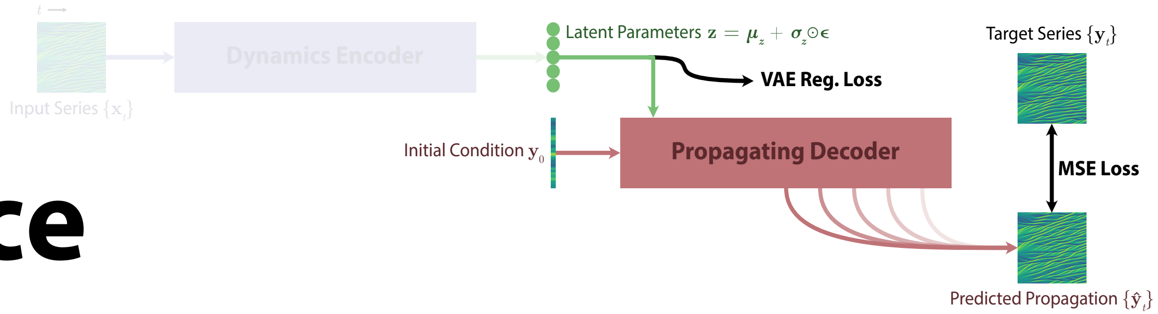




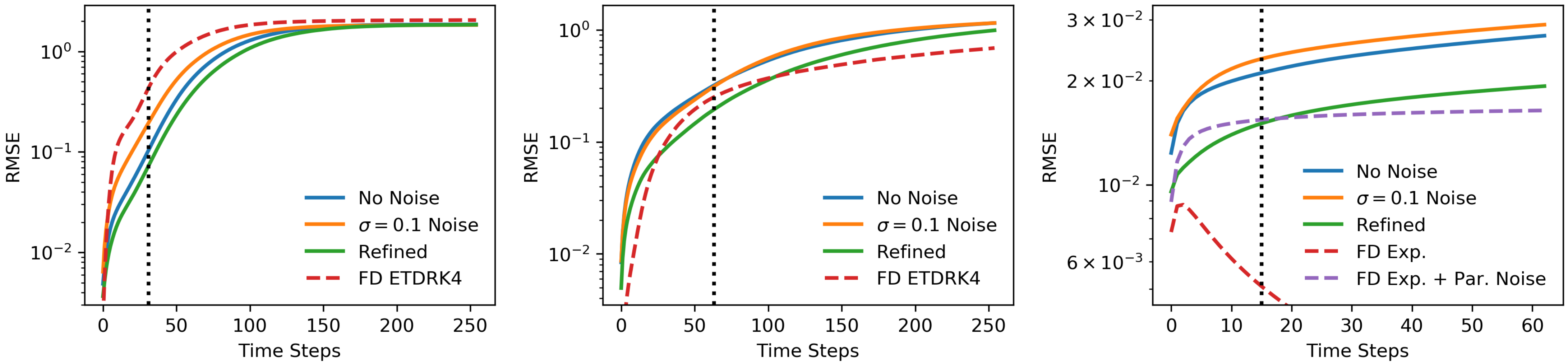
(a) KS, Test Example
(b) KS, Model Prediction
(c) KS, Model Error
(d) NS, Test Example
(e) NS, Model Prediction
(f) NS, Model Error
(g) CD, Test Example
(h) CD, Model Prediction
(i) CD, Model Error

arXiv:1907.06011

Decoder Prediction Performance



Performance comparable to finite-difference method with same space discretization and time step.



(a) Kuramoto–Sivashinsky (b) Nonlinear Schrödinger (c) Convection–Diffusion

arXiv:1907.06011

Summary and Conclusions

- Method achieved successful interpretable parameter extraction on: convection–diffusion, nonlinear Schrödinger, and Kuramoto–Sivashinsky.
- We observed robustness to noise.*
- **Future work:** Test on experimental datasets and applications, e.g. fluid imaging, chemical/biological systems.

➢ Unsupervised learning can aid in our understanding of physical systems.

➢ We can optimize the trade-off between flexibility and interpretability using physics-informed deep learning architectures.

➢ Code available: https://github.com/peterparity/PDE-VAE-pytorch

arXiv:1907.06011

Acknowledgments

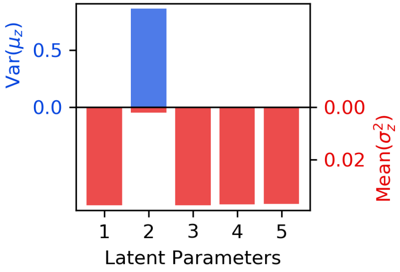
Funding: MIT, NDSEG Fellowship, DARPA

Discussions: Rumen Dangovski, Li Jing, Jason Fleischer

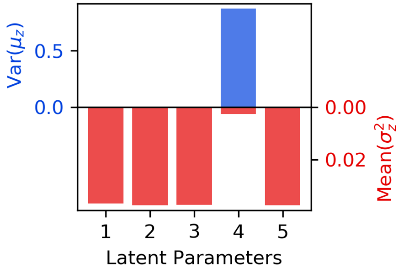
Samuel Kim

Marin Soljačić

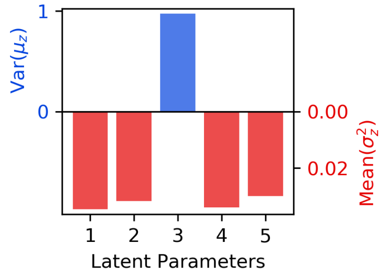
Raw Parameter Extraction



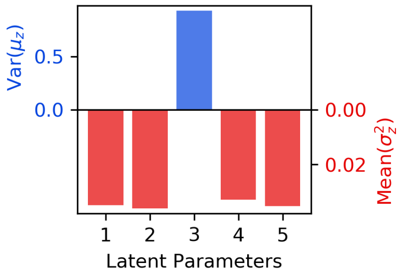
(a) KS, No Noise



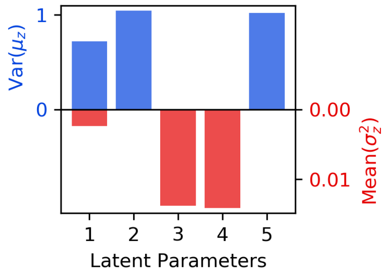
(b) KS, $\sigma = 0.1$ Noise



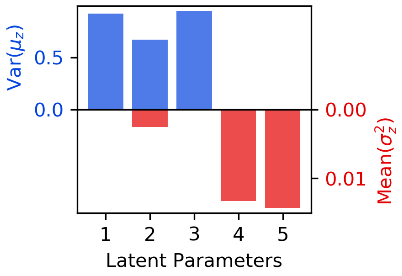
(c) NS, No Noise



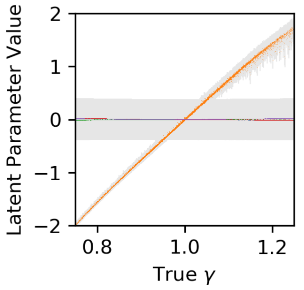
(d) NS, $\sigma = 0.1$ Noise



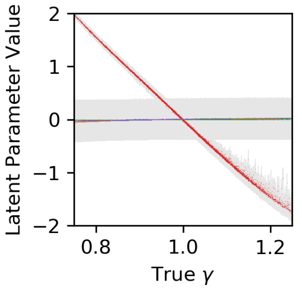
(e) CD, No Noise



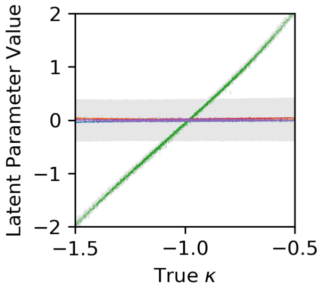
(f) CD, $\sigma = 0.1$ Noise



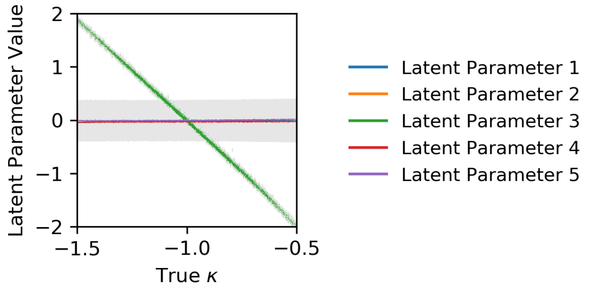
(a) KS, No Noise



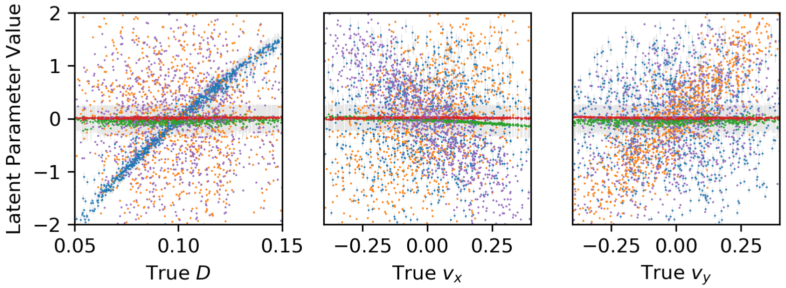
(b) KS, $\sigma = 0.1$ Noise



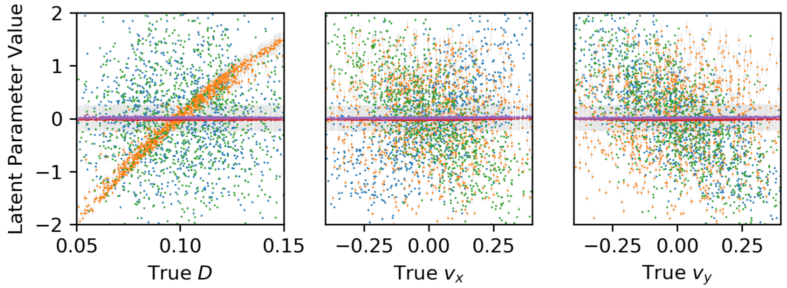
(c) NS, No Noise



(d) NS, $\sigma = 0.1$ Noise



(e) CD, No Noise



(f) CD, $\sigma = 0.1$ Noise

arXiv:1907.06011

VAE Regularization

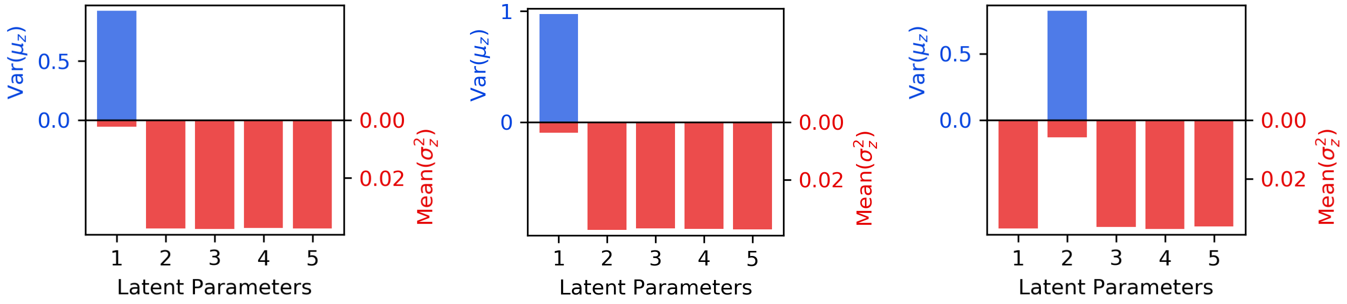
$$
\begin{aligned}
\mathbb{E}_{p_D(\mathbf{x})}[D_{\mathrm{KL}}(q(\mathbf{z}|\mathbf{x}) \parallel p(\mathbf{z}))] = \\
& D_{\mathrm{KL}}(q(\mathbf{z},\mathbf{x}) \parallel q(\mathbf{z})\, p_D(\mathbf{x})) \\
& + D_{\mathrm{KL}}(q(\mathbf{z}) \parallel \prod_i q(z_i)) \\
& + \sum_i D_{\mathrm{KL}}(q(z_i) \parallel p(z_i))
\end{aligned}
$$

1. Parsimony
2. Statistical independence
3. Gaussian prior

arXiv:1907.06011

Data Sparsity

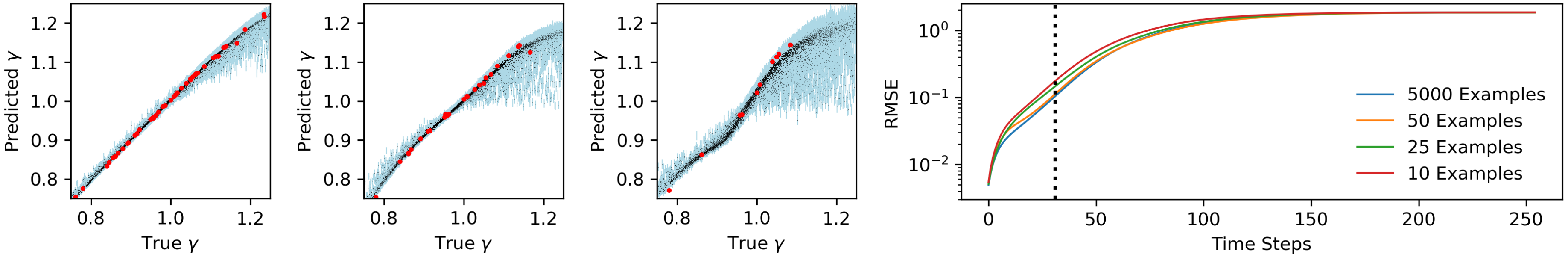
Trained on only 50, 25, or 10 examples from Kuramoto–Sivashinsky dataset.



(a) Identification: 50 Examples

(b) Identification: 25 Examples

(c) Identification: 10 Examples



(d) Extraction: 50 Examples

(e) Extraction: 25 Examples

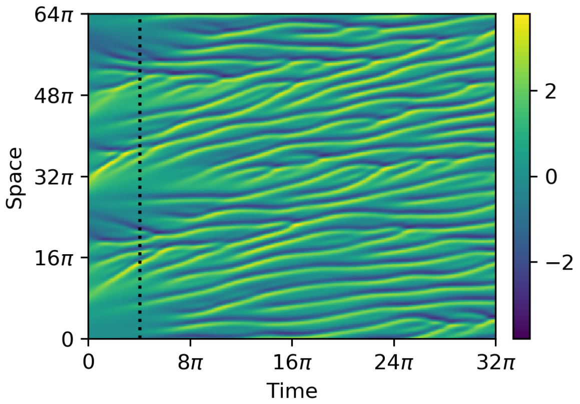
(f) Extraction: 10 Examples

(g) Model Prediction

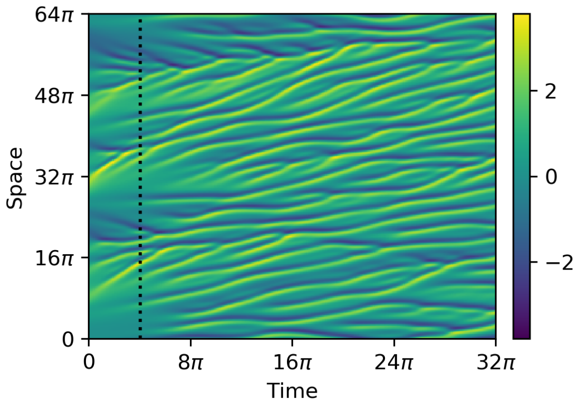
arXiv:1907.06011

Evaluation with alternative B.C.s

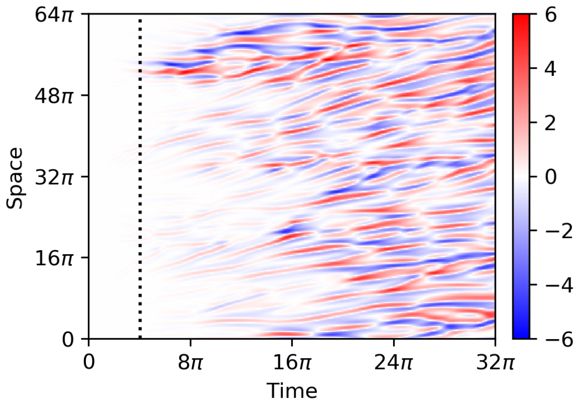
Periodic Boundaries



(a) KS, Test Example

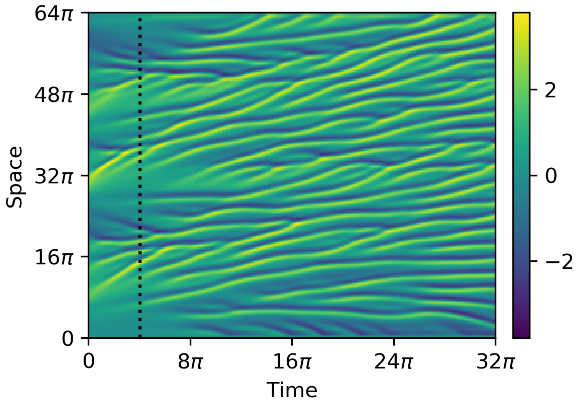


(b) KS, Model Prediction

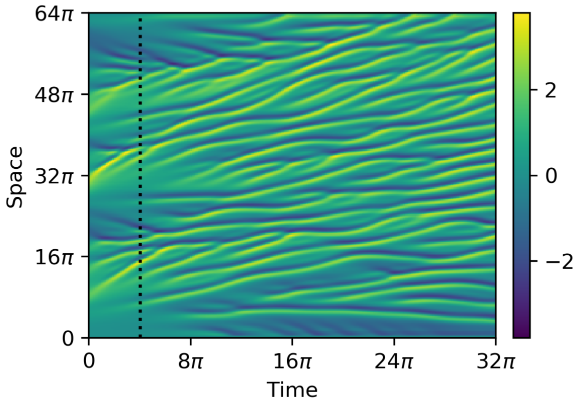


(c) KS, Model Error

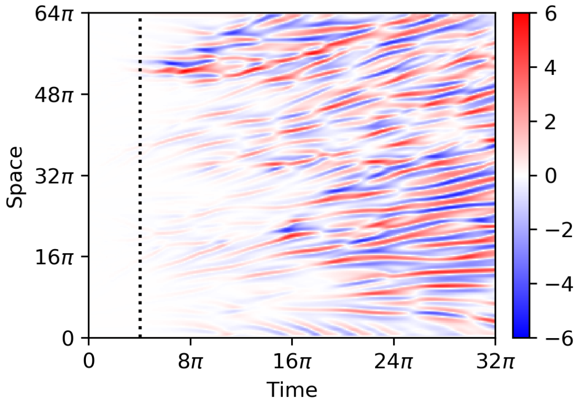
Dirichlet Boundaries



(a) Test Example



(b) Model Prediction



(c) Model Error

arXiv:1907.06011