CS230

Learning reduced order models for fluid flow phenomena from high-fidelity simulation data

Category: Deep Autoencoders

Mark Benjamin
Department of Mechanical Engineering
Stanford University
markben@stanford.edu

Konrad Goc
Department of Aeronautics and Astronautics
Stanford University
kgoc@stanford.edu

Abstract

In this work, we use a combined deep autoencoder and a sparse regression algorithm (SINDy Autoencoders) Champion et al. [2019] to discover parsimonious models governing dynamical systems. The autoencoder enables the discovery of a lower dimensional representation of the high-fidelity simulation data, and the SINDy regression algorithm Rudy et al. [2017] looks for a sparse model made up of (in this case) a library $\Theta(\vec{z})$ of polynomial basis functions to represent the model *in this* lower dimensional space. Training of the autoencoder and the regression model occurs simultaneously, thereby ensuring that the lower dimensional space that is discovered is one that permits a sparse dynamical model.

1 Introduction

Given high-fidelity spatio-temporal data from solutions to partial differential equations governing fluid flow phenomena, we seek to discover parsimonious reduced order models that describe the dynamics governing the system. This approach will reduce the computational cost of high-fidelity numerical simulations.

The SINDy loss function is given by

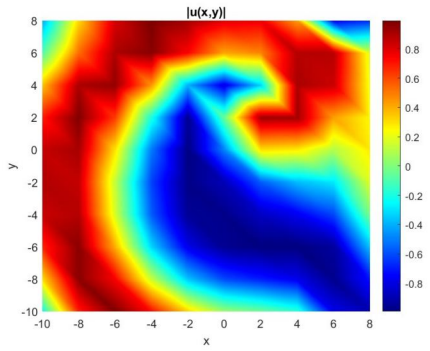
$$\|\mathbf{x} - \psi(\mathbf{z})\|_2^2 + \lambda_1 \left\|\dot{\mathbf{x}} - (\nabla_{\mathbf{z}}\psi(\mathbf{z}))\left(\boldsymbol{\Theta}\left(\mathbf{z}^T\right)\boldsymbol{\Xi}\right)\right\|_2^2 + \lambda_2 \left\|(\nabla_{\mathbf{x}}\mathbf{z})\,\dot{\mathbf{x}} - \boldsymbol{\Theta}\left(\mathbf{z}^T\right)\boldsymbol{\Xi}\right\|_2^2 + \lambda_3\|\boldsymbol{\Xi}\|_1 \quad (1)$$

where the first term is the reconstruction loss for the autoencoder, the second and third are the SINDy losses in the input and latent spaces respectively, and the final term is added for regularization. $\mathbf{x}$ is the input data in physical space, $\mathbf{z}$ is the corresponding encoding, Θ is the matrix of basis functions, and Ξ is the matrix of coefficients. λ_i are regularization coefficients.

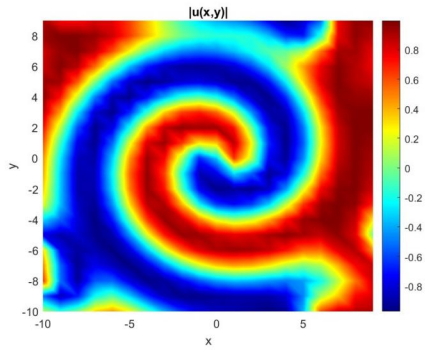
2 Training data

The training data is obtained by a numerical simulation of the reaction diffusion equation, using a Fourier discretization in space, and a Runge-Kutta time integrator. While this is a non-linear, second-order PDE in two dimensions, it exhibits a spiral-like long term behavior that is described by a simple harmonic oscillator. This a priori knowledge of a reduced representation allows us to test the performance of SINDy.

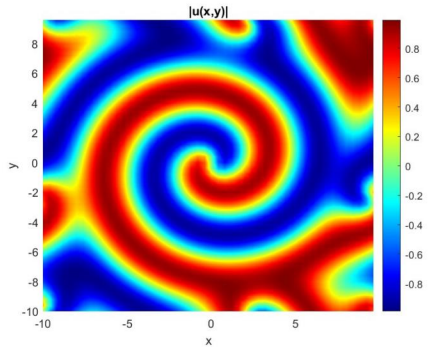
CS230: Deep Learning, Spring 2022, Stanford University, CA. (LateX template borrowed from NIPS 2017.)



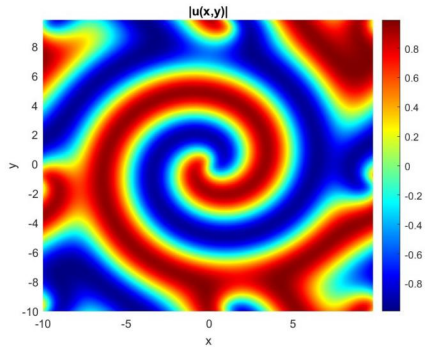
(a) Coarse 10×10 grid solution.



(b) Medium 20×20 grid solution.



(c) Fine 50×50 grid solution.



(d) Extra fine 100×100 grid solution.

Figure 1: Solutions obtained using a spectral Reaction-Diffusion solver on grids of four different levels of refinement, a "coarse" 10×10 mesh, a "medium" 20×20 mesh, a "fine" 50×50 mesh, and an "extra fine" 100×100 mesh.

Table 1: Details of SINDy-Autoencoder model

Model inputs	$\vec{x}(t) \in \mathbb{R}^N$
Model outputs	Coefficient matrix Ξ
Model hyperparameters	Latent dimension size d
	Kernel polynomial order p
	Autoencoder architecture

The governing equations for the reaction-diffusion problem are as follows:

$$u_t = \left(1 - \left(u^2 + v^2\right)\right) u + \beta \left(u^2 + v^2\right) v + d_1 \left(u_{xx} + u_{yy}\right) \tag{2}$$

$$v_t = -\beta \left(u^2 + v^2\right) u + \left(1 - \left(u^2 + v^2\right)\right) v + d_2 \left(v_{xx} + v_{yy}\right) \tag{3}$$

where u and v are the velocities in the x and y dimension respectively, β is a reaction rate parameter, and d_1 and d_2 are diffusivity coefficients.

We first evaluate the model on a high resolution mesh, with 10000 nodes, which is sufficient to resolve all scales of motion. The inputs to the model consist of the values of velocities in two dimensions (u and v) at all spatial locations, in a vector $\vec{x}(t) \in \mathbb{R}^{20000}$. Next, we run a coarse simulation of the same problem, reducing the spatial resolution by a factor of 4. Under-resolved numerical simulations are commonplace in fluid dynamics, owing to the prohibitive computational costs associated with direct simulation; thus, the impact of numerical error on discovered dynamics is of interest.

Table 2: Error in the Modeled Dynamics Relative to the True Dynamics

10×10 Mesh	1.000000
20×20 Mesh	1.105174
50×50 Mesh	0.005814
100×100 Mesh	0.002132

3 Approach and results

3.1 Grid Resolution Sensitivity

Our initial testing of the resulting reduced order models is described in Figure 3.1. We have carried out the coefficient learning process on four numerical solutions to the reaction-diffusion equation, ranging from a coarse to an extra fine one. Figure 1 shows the differences between these solutions, with the extra fine grid solution being a more accurate representation of the true dynamics in the governing PDEs. We find that the more accurate the data used as a starting point (i.e. from the fine or extra fine mesh), the better the simulated dynamics represent the true dynamics of the system. Interestingly, we also find that for the extra fine mesh solution, SINDy identifies only two active terms in the dynamics of the system, while on the coarse mesh, the autoencoder fails to identify an relevant terms, owing to the poor quality numerical solution generated on the coarse grid. It is difficult to identify relevant physics when the solution to the governing PDE is poor. This can be seen in Equations 4 through 7.

$$\begin{aligned} \dot{z_1} &= 0 \\ \dot{z_2} &= 0 \end{aligned} \tag{4}$$

$$\begin{aligned} \dot{z_1} &= 0 \\ \dot{z_2} &= 0.14 z_2^2 \end{aligned} \tag{5}$$

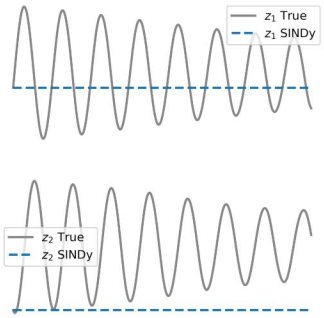
$$\begin{aligned} \dot{z_1} &= -1.04 sin(z_2) \\ \dot{z_2} &= 0.84 z_1 \end{aligned} \tag{6}$$

$$\begin{aligned} \dot{z_1} &= -0.85 z_2 \\ \dot{z_2} &= 0.97 z_1 \end{aligned} \tag{7}$$

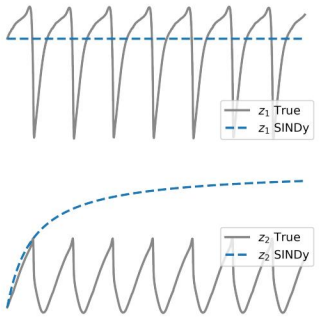
3.2 Sensitivity to Basis Functions

We further evaluated the sensitivity to the predictions of the Autoencoder + SINDy to the choice of terms included in the basis function set. We considered modulating both the polynomial degree and the presence of sinusoids in the basis set. In the grid convergence study, the maximum allowable polynomial degree was set to 3. Because the modeled dynamics never included terms of higher degree than 2 in that experiment, we decided to explore the sensitivity to the inclusion of the sinusoids on the fine grid, which had a sine in its predicted dyanamics. Figure 3b shows the results of this experiment. We find that on the given grid, the autoencoder + SINDy is unable to adequately model the dynamics of the system when we prevent it from using sinusoids in the basis function set.

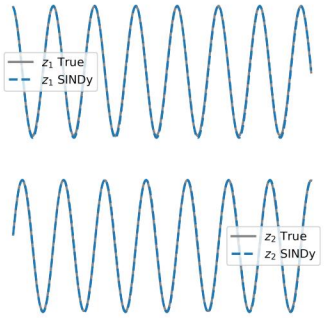
The reason for this may be thought of along the following lines: the added numerical noise when solving for the 50×50 solution introduces small values of higher order coefficients that are swept away in the sequential-thresholding performed by the least-squares to promote sparsity. Since the dynamics of the system are easily approximated by a cyclical waveform, a sine function is a good candidate to represent the effects of those higher order terms that were set to zero. However, as one refines the grid further, the true dynamics emerge, causing the sine term to vanish.



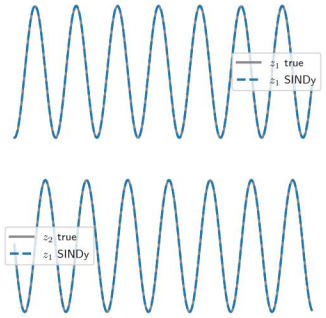
(a) Dynamics predicted by SINDy + Autoencoder on the coarse 10×10 grid.



(b) Medium 20×20 grid solution.



(c) Fine 50×50 grid solution.



(d) Extra fine 100×100 grid solution.

Figure 2: Autoencoder + SINDy dynamics predicted on a "coarse" 10×10 mesh up to an "extra fine" 100×100 mesh. The fidelity of the prediction is improved significantly when more accurate training data is used as a starting point for the learning.

4 Limitations

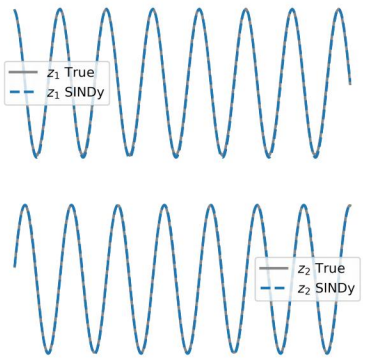
Having established that the algorithm recovers the expect sparsity pattern in the reaction-diffusion problem, we next consider a problem for which a sparse representation is not known a priori: the 2D advection equation, given by

$$\partial_t u = -c_x \partial_x u - c_y \partial_y u \tag{8}$$

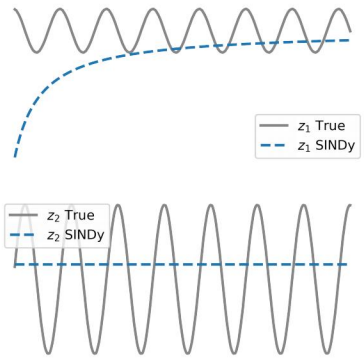
A snapshot of the solution is shown in Figure 4. Investigations show that the SINDy algorithm is unable to produce a truly sparse representation of this system, requiring $\mathcal{O}(100)$ coefficients to represent the dynamics in the encoded space. In other words, the discovered reduced dimension does not provide any sparsity. Thus, we may conclude that the effectiveness of the algorithm is limited by the inherent physics of the problem. These results are consistently observed across a range of hyperparameters.

5 Conclusions

In this project, we have examined the performance of the SINDy algorithm for discovering reduced representations of dynamical systems. We first consider a problem for which the solution permits a



(a) Dynamics predicted by SINDy + Autoencoder on the coarse 50×50 grid, including the sinusoids in the basis function set.



(b) Dynamics predicted by SINDy + Autoencoder on the coarse 50×50 grid, excluding the sinusoids in the basis function set.

Figure 3: Autoencoder + SINDy dynamics predicted on a "fine" 50×50 mesh using basis set that both includes and excludes sines. The fidelity of the prediction is improved significantly when including the sine term in the function basis at this grid resolution.

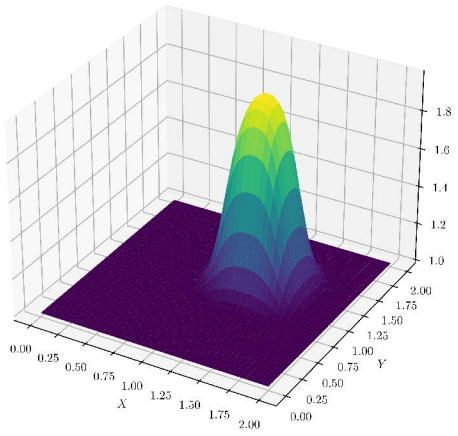


Figure 4: Solution of the advection equation at $t = 0.4$. The hump is initialized at the bottom left of the grid and is diagonally advected across the domain.

reduced representation: the reaction-diffusion equation. We evaluate the performance of the algorithm on a variety of numerical discretizations, and find that the sparse representation degrades rapidly when numerical artifacts seep into the solution. We attribute much of the poor prediction of dynamics at these lower resolutions to the nature of the algorithm; in particular, the ad hoc thresholding that is periodically applied during the training process to remove coefficients close to zero. One future avenue of exploration might be to consider the threshold used as a parameter, rather than a hyperparameter, and include it as part of the optimization loop. Finally, we also evaluate the model's performance on a problem with no known sparse representation - the advection equation - and show that an accurate solution reconstruction requires a dense matrix of coefficents.

5.1 Contributions

K.G. performed the grid resolution and basis function sensitivity assessments. M.B. evaluated the limitations of the method and performed the literature survey. Both authors contributed equally to the writing of the summary and PowerPoint presentation.

References

Kathleen Champion, Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116 (45):22445–22451, 2019. doi: 10.1073/pnas.1906995116.

Samuel H. Rudy, Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Data-driven discovery of partial differential equations. *Science Advances*, 3(4):e1602614, 2017. doi: 10.1126/sciadv.1602614. URL `https://www.science.org/doi/abs/10.1126/sciadv.1602614`.