Data-driven discovery of partial differential equations

Samuel H. Rudy[1]*, Steven L. Brunton[2], Joshua L. Proctor[3], and J. Nathan Kutz[1]
[1] *Department of Applied Mathematics, University of Washington, Seattle, WA. 98195*
[2] *Department of Mechanical Engineering, University of Washington, Seattle, WA. 98195 and*
[3] *Institute for Disease Modeling, 3150 139th Ave SE, Bellevue, WA 98005*

(Dated: October 23, 2016)

We propose a sparse regression method capable of discovering the governing partial differential equation(s) of a given system by time series measurements in the spatial domain. The regression framework relies on sparsity promoting techniques to select the nonlinear and partial derivative terms of the governing equations that most accurately represent the data, bypassing a combinatorially large search through all possible candidate models. The method balances model complexity and regression accuracy by selecting a parsimonious model via Pareto analysis. Time series measurements can be made in an Eulerian framework where the sensors are fixed spatially, or in a Lagrangian framework where the sensors move with the dynamics. The method is computationally efficient, robust, and is demonstrated to work on a variety of canonical problems spanning a number of scientific domains including Navier-Stokes, the quantum harmonic oscillator, and the diffusion equation. Moreover, the method is capable of disambiguating between potentially non-unique dynamical terms by using multiple time series taken with different initial data. Thus for a traveling wave, the method can distinguish between a linear wave equation or the Korteweg-deVries equation, for instance. The method provides a promising new technique for discovering governing equations and physical laws in parametrized spatio-temporal systems where first-principles derivations are intractable.

PACS numbers:

I. INTRODUCTION

Data-driven discovery methods, which have been enabled in the last decade by the plummeting cost of sensors, data storage and computational resources, are having a transformative impact on the sciences, facilitating a variety of innovations for characterizing high dimensional data generated from experiments. Less well understood is how to uncover underlying physical laws and/or governing equations from time series data that exhibit spatio-temporal activity. Traditional theoretical methods for deriving the underlying partial differential equations (PDEs) are rooted in conservation laws, physical principles and/or phenomenological behaviors. These first principle derivations lead to many of the canonical models ubiquitous in physics, engineering, and the biological sciences. However, there remain many complex systems that have eluded quantitative analytic descriptions or even characterization of a suitable choice of variables (e.g. neuroscience, power grid, epidemiology, finance, ecology, etc). We propose an alternative method to derive governing equations based solely on time series data collected at a fixed number of spatial locations. Using innovations in sparse regression, we discover the terms of the governing PDE that most accurately represent the data from a large library of potential candidate functions. Importantly, measurements can be made in an Eulerian framework where the sensors are fixed spatially, or in a Lagrangian framework where the sensors move with the dynamics. We demonstrate the success of the method by re-discovering a broad range of physical laws from solely time series data.

Methods for data-driven discovery of dynamical systems [1] include equation-free modeling [2], artificial neural networks [3], nonlinear regression [4], empirical dynamic modeling [5, 6], normal form identification [7], nonlinear Laplacian spectral analysis [8], modeling emergent behavior [9], and automated inference of dynamics [10–12]. In this series of developments, seminal contributions leveraging symbolic regression and an evolutionary algorithm [13, 14] were capable of directly determining nonlinear dynamical system from data. More recently, sparsity [15] has been used to robustly determine, in a highly efficient computational manner, the governing dynamical system [16, 17]. Both the evolutionary [14] and sparse [16] symbolic regression methods avoid overfitting by selecting parsimonious models that balance model accuracy with complexity via Pareto analysis.

The method we present is able to select, from a large library, the correct linear, nonlinear, and spatial derivative terms, resulting in the identification of PDEs from data. Only those terms that are most informative about the dynamics are selected as part of the discovered PDE. The innovation presented here is critically important since it efficiently handles spatio-temporal data which is a fundamental characteristic of many canonical models. Previous sparsity-promoting methods are able to identify ordinary differential equations (ODEs) from data, but are not able to handle spatio-temporal data, nor high-dimensional measurements [16]. Our novel methodology has a number of advantageous practical characteristics: measurements can be collected in either a fixed or moving frame (Eulerian or Lagrangian) allowing for a broad application to a variety of experimental data; the algorithm

*Electronic address: shrudy@uw.edu

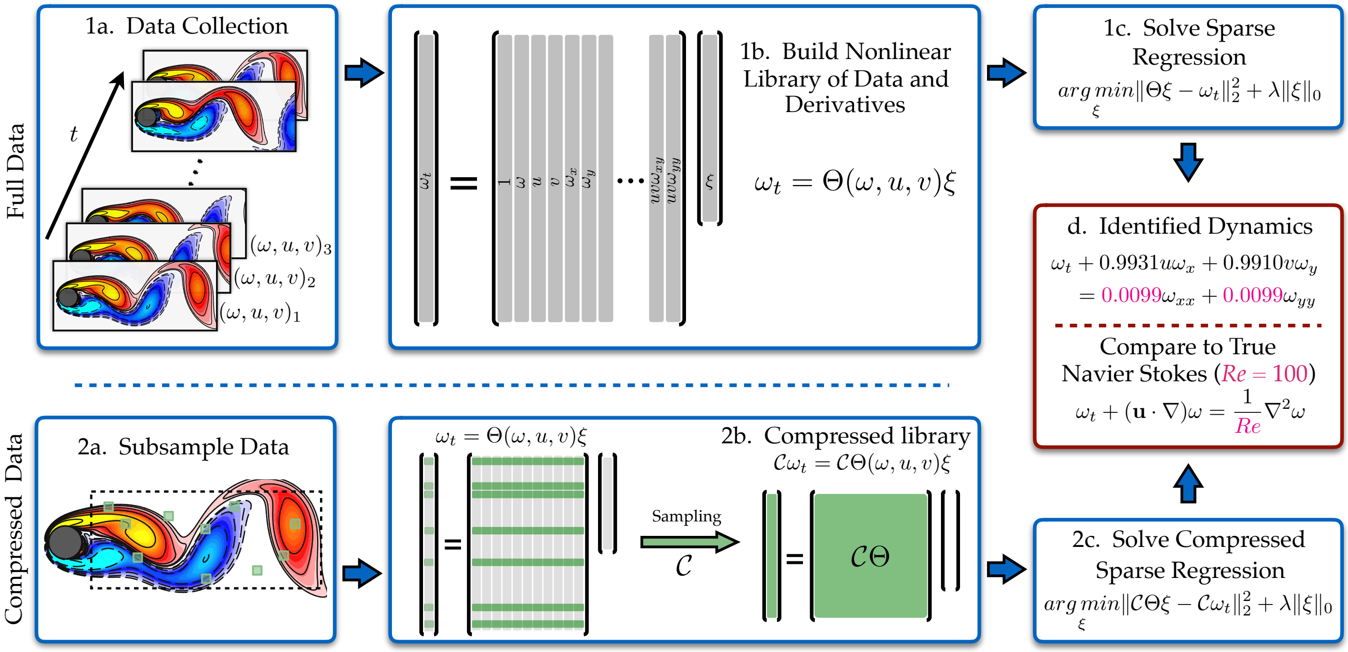

Full Data
1a. Data Collection
t
$(\omega,u,v)_3$
$(\omega,u,v)_2$
$(\omega,u,v)_1$
1b. Build Nonlinear Library of Data and Derivatives
ω_t
1 ω u v ω_x ω_y $\cdots$ $uv\omega_{xy}$ $uv\omega_{yy}$
ξ
$\omega_t = \Theta(\omega,u,v)\xi$
1c. Solve Sparse Regression
$\underset{\xi}{arg\,min}\|\Theta\xi-\omega_t\|_2^2+\lambda\|\xi\|_0$
d. Identified Dynamics
$\omega_t + 0.9931u\omega_x + 0.9910v\omega_y$
$= 0.0099\omega_{xx} + 0.0099\omega_{yy}$
Compare to True Navier Stokes ($Re = 100$)
$\omega_t + (\mathbf{u}\cdot\nabla)\omega = \frac{1}{Re}\nabla^2\omega$
Compressed Data
2a. Subsample Data
$\omega_t = \Theta(\omega,u,v)\xi$
2b. Compressed library
$\mathcal{C}\omega_t = \mathcal{C}\Theta(\omega,u,v)\xi$
Sampling
$\mathcal{C}$
$\mathcal{C}\Theta$
2c. Solve Compressed Sparse Regression
$\underset{\xi}{arg\,min}\|\mathcal{C}\Theta\xi-\mathcal{C}\omega_t\|_2^2+\lambda\|\xi\|_0$

FIG. 1: Steps in the PDE functional identification of nonlinear dynamics (PDE-FIND) algorithm, applied to infer the Navier-Stokes equations from data. **1a.** Data is collected as snapshots of a solution to a PDE. **1b.** Numerical derivatives are taken and data is compiled into a large matrix $\boldsymbol{\Theta}$, incorporating candidate terms for the PDE. **1c.** Sparse regressions is used to identify active terms in the PDE. **2a.** For large datasets, sparse sampling may be used to reduce the size of the problem. **2b.** Subsampling the dataset is equivalent to taking a subset of rows from the linear system in (5). **2c.** An identical sparse regression problem is formed but with fewer rows. **d.** Active terms in ξ are synthesized into a PDE.

can efficiently handle high-dimensional data through innovative sampling strategies. The algorithm, *PDE functional identification of nonlinear dynamics* (PDE-FIND), is applied to a wide-range of canonical models.

II. RESULTS

We consider a parametrized and nonlinear PDE of the general form

$$u_t = N(u, u_x, u_{xx}, \cdots, x, \mu), \tag{1}$$

where the subscripts denote partial differentiation in either time or space, and $N(\cdot)$ is an unknown right-hand side that is generally a nonlinear function of $u(x,t)$, its derivatives, and parameters in μ. Our objective is to construct $N(\cdot)$ given time series measurements of the system at a fixed number of spatial locations in x. A key assumption is that the function $N(\cdot)$ consists of only a few terms, making the functional form sparse relative to the large space of possible contributing terms. As an example, Burgers' equation ($N = -uu_x + \mu u_{xx}$) and the harmonic oscillator ($N = -i\mu x^2 u - i\hbar u_{xx}/2$) each have two terms. Thus, given the large collection of candidate terms for constructing PDEs, we utilize sparse regression methodologies to determine *which* right-hand-side terms are contributing to the dynamics without an intractable (np-hard) combinatorial brute-force search across all possible term combinations.

Upon discretization, the right hand side of (1) can be expressed as a function of $\mathbf{U}$, which is the discrete version of $u(x,t)$ and its derivatives, through the matrix $\boldsymbol{\Theta}(\mathbf{U},\mathbf{Q})$ where the column vector $\mathbf{Q}$ contains any additional input terms to the right hand side. Each column of the library $\boldsymbol{\Theta}(\mathbf{U},\mathbf{Q})$ corresponds to a specific candidate term for the governing equation, as shown in figure 1 panel 1b. The PDE evolution can be expressed in this library as follows:

$$\mathbf{U}_t = \boldsymbol{\Theta}(\mathbf{U},\mathbf{Q})\xi. \tag{2}$$

Each nonzero entry in ξ corresponds to a term in the PDE, and for canonical PDEs, the vector ξ is *sparse*, meaning that only a few terms are active. The Materials and Methods section shows explicitly how to construct $\boldsymbol{\Theta}(\mathbf{U},\mathbf{Q})$ and solve for the vector ξ, thus identifying the terms in the PDE.

Discovering the Navier-Stokes Equations

Figure 1 demonstrates the algorithmic procedure for successfully identifying the correct PDE dynamics for a given set of measurements from a physical system. Specifically, fluid flow around a cylinder is simulated at a given Reynolds, and measurements of the vorticity can be densely or sparsely sampled in order to correctly reconstruct the well-known Navier-Stokes equations. Remarkably, the coefficients of the PDE and Reynolds number are identified within a fraction of a percent accuracy. This figure represents our innovative mathematical structure that combines sparse regression, a library of potential functional forms, and parsimonious model selection.

Figure 1 also demonstrates that for large datasets, such as those generated from 2D and 3D problems, PDE-FIND can be effectively used on subsampled data. This distinction is fundamentally important since full-state measurements are often computationally and experimentally prohibitive to collect, and they may also make the regression needlessly expensive. We randomly select a set of spatial points and uniformly subsample in time, resulting in the use of only a fraction of the dataset. Mathematically, this amounts to ignoring a fraction of the rows in the linear system $\mathbf{U}_t = \mathbf{\Theta}(\mathbf{U}, \mathbf{Q})\xi$ as illustrated in Figure 1 panels 2a and 2b. Although we only use a small fraction of the spatial points in the linear system, nearby points are needed to evaluate the derivative terms in the library. The derivatives are computed using a small number of spatially localized points near each measurement position via polynomial interpolation. Therefore, while subsampling uses only a small fraction of the points in the regression, we are using local information around each measurement.

Previous sparse identification algorithms [16] faced a number of challenges: they were not able to handle subsampled spatial data and the algorithm did not scale well to high-dimensional measurements. Standard model reduction techniques such as proper orthogonal decomposition (POD) were used to overcome the high-dimensional measurements, allowing for a lower order ODE model to be constructed on energetic POD modes. This procedure resembles the standard Galerkin projection onto POD modes [18]. In contrast, the PDE-FIND algorithm identifies a PDE *directly* from subsampled measurement data.

Sensors Moving with the Dynamics

As a second demonstration of the method, we consider one of the fundamental results of the early 20th century concerning the relationship between random walks (Brownian motion) and diffusion. The theoretical connection between these two was first made by Einstein in 1905 [19] and was part of the *Annus Mirablis papers* which lay the foundations of modern physics. We use the method proposed here to sample the movement of a random walker, which is effectively a Lagrangian measurement coordinate, in order to verify that it can reproduce the well-known diffusion equation. By biasing the random walk, we can also produce the generalization of advection-diffusion in one-dimension. Figure 2 shows the success of the method in identifying the correct diffusion model from a random walk trajectory. Given a sufficiently long time series with high enough resolution, the method produces the heat equation which describes the temporal evolution of the probability distribution function.

This example is important in that the measurements are now moving with the dynamics of the underlying governing equations. Specifically, measurements are made in the frame of the random walker. For many physical and engineering systems, fixed measurement locations may be impractical to achieve. This is relevant for applications such as ocean monitoring where the change in spatial location of buoys due to ocean currents can be informative about the underlying dynamical properties of the system. Such systems represent a Lagrangian framework for sensor placement and data-driven discovery of dynamics.

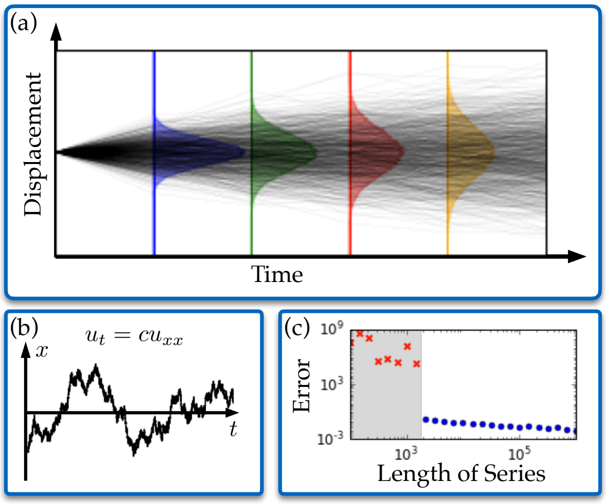


FIG. 2: Inferring the diffusion equation from a single Brownian motion. (**a**) Time series is broken into many short random walks that are used to construct histograms of the displacement. (**b**) The Brownian motion trajectory, following the diffusion equation. (**c**) Parameter error ($\|\xi^* - \hat{\xi}\|_1$) vs. length of known time series. Blue symbols correspond to correct identification of the structure of the diffusion model, $u_t = cu_{xx}$.

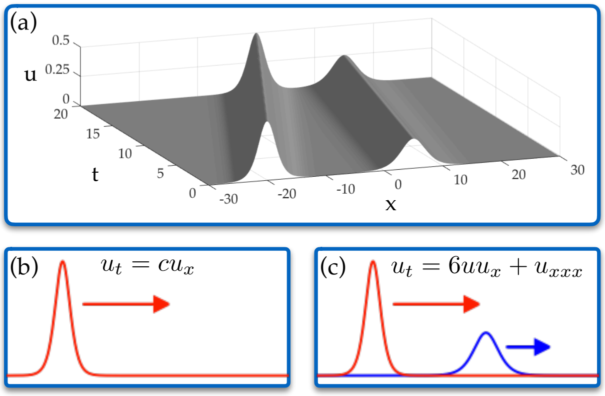


FIG. 3: Inferring nonlinearity via observing solutions at multiple amplitudes. (**a**) An example 2-soliton solution to the KdV equation. (**b**) Applying our method to a single soliton solution determines that it solves the standard advection equation. (**c**) Looking at two completely separate solutions reveals nonlinearity.

Disambiguating Dynamical Systems

A third canonical example is the KdV equation modeling the unidirectional propagation of small-amplitude, long water waves or shallow-water waves. Discovered first by Boussinesq in 1877 and later developed by Korteweg

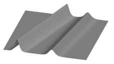
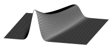
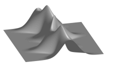
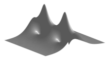
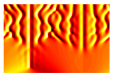
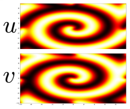
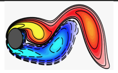
PDE	Form	Error (no noise, noise)	Discretization
 KdV	$u_t + 6uu_x + u_{xxx} = 0$	1%±0.2%, 7%±5%	$x \in [-30, 30], n{=}512,\ t \in [0, 20], m{=}201$
 Burgers	$u_t + uu_x - \epsilon u_{xx} = 0$	0.15%±0.06%, 0.8%±0.6%	$x \in [-8, 8], n{=}256,\ t \in [0, 10], m{=}101$
 Schrödinger	$iu_t + \frac{1}{2}u_{xx} - \frac{x^2}{2}u = 0$	0.25%±0.01%, 10%±7%	$x \in [-7.5, 7.5], n{=}512,\ t \in [0, 10], m{=}401$
 NLS	$iu_t + \frac{1}{2}u_{xx} + \lvert u\rvert^2 u = 0$	0.05%±0.01%, 3%±1%	$x \in [-5, 5], n{=}512,\ t \in [0, \pi], m{=}501$
 KS	$u_t + uu_x + u_{xx} + u_{xxxx} = 0$	1.3%±1.3%, 52%±1.4%	$x \in [0, 100], n{=}1024,\ t \in [0, 100], m{=}251$
 Reaction Diffusion	$u_t = 0.1\nabla^2 u + \lambda(A)u - \omega(A)v$ $v_t = 0.1\nabla^2 v + \omega(A)u + \lambda(A)v$ $A^2 = u^2 + v^2, \omega = -\beta A^2, \lambda = 1 - A^2$	0.02% ± 0.01%, 3.8% ± 2.4%	$x, y \in [-10, 10], n{=}256,\ t \in [0, 10], m{=}201$ subsample 1.14%
 Navier Stokes	$\omega_t + (\mathbf{u} \cdot \nabla)\omega = \frac{1}{Re}\nabla^2\omega$	1% ± 0.2% , 7% ± 6%	$x \in [0, 9], n_x{=}449,\ y \in [0, 4], n_y{=}199,$ $t \in [0, 30], m{=}151$, subsample 2.22%

TABLE I: Summary of regression results for a wide range of canonical modes of mathematical physics. In each example, the correct model structure is identified using PDE-FIND. The spatial and temporal sampling used for the regression is given along with the error produced in the parameters of the model for both no noise and 1% noise. In the reaction-diffusion (RD) system, 0.5% noise is used. For Navier Stokes and Reaction Diffusion, the percent of data used in subsampling is also given.

and deVries in 1895, it was one of the earliest models known to have soliton solutions. One potential viewpoint of the equation is as a dispersive regularization of Burgers' equation. The KdV evolution is given by

$$u_t + 6uu_x + u_{xxx} = 0\,, \tag{3}$$

with soliton solutions $u(x,t) = (c/2)\,\mathrm{sech}^2[(\sqrt{c}/2)(x - ct - x_0)]$ which propagate with a speed proportional to their amplitude c. Interestingly, if one observes a single propagating soliton, it would be indistinguishable from a solution to the one-way wave equation $u_t + cu_x = 0$. As such, it presents a challenge to PDE-FIND, which would select the sparsest representation, in this case the one-way wave equation. This ambiguity in selecting the correct PDE is rectified by constructing time series data for more than a single initial amplitude. Figure 3 demonstrates the evolution of two KdV solitons of differing amplitudes, which enables the unique determination of the governing PDE (3).

Discovery for Canonical PDE Models

Table I applies the methodology proposed to a wide range of canonical models from mathematical physics and engineering sciences. The PDEs selected represent a wide range of physical systems, displaying both Hamiltonian (conservative) dynamics and dissipative nonlinear dynamics along with periodic to chaotic behavior. Aside from the quantum oscillator (3rd row), all the dynamics observed are strongly nonlinear. Remarkably, the method is able to discover each physical system even if significantly subsampled spatially. The space and time sampling required, along with the accuracy in recovering the PDE parameters with and without noise, are detailed in the Table. This highlights the broad applicability of the method and the success of the technique in discovering governing PDEs. Moreover, it demonstrates that a wide range of distinct, physical observations can be captured by the regression framework, thus allowing the method to be broadly applied across the sciences where spatio-temporal activity dominates.

III. DISCUSSION

PDE-FIND is a viable, data-driven tool for modern applications where first-principles derivations may be intractable (e.g. neuroscience, epidemiology, dynamical networks), but where new data recordings and sensor technologies are revolutionizing our understanding of physical and/or biophysical processes on spatial domains. To our knowledge, this is the first data-driven regression technique that explicitly accounts for spatial derivatives in discovering physical laws, thus significantly broadening the applicability to a wide-variety of complex systems. Traditional approaches, such as reducing the model to a local linear embedding based on collected data, fall short in achieving a fundamental goal of of research scientists: to construct a nonlinear model from observations which characterizes the observed dynamics and generalizes to unsampled areas of parameter space. For instance, we can discover the Navier-Stokes equations at $Re = 100$ and use this knowledge to accurately simulate a fully turbulent system at $Re = 10000$ where no data was collected. This represents a significant paradigm shift when compared with most machine learning architectures where accurate predictions can only be made near parameter regimes where the data was sampled.

The success of this methodology suggests that many concepts from statistical learning can be integrated with traditional scientific computing and dynamical systems theory to discover dynamical models from data. This integration of nonlinear dynamics and machine learning opens the door for principled versus heuristic methods for model construction, nonlinear control strategies, and sensor placement techniques. Moreover, these new model identification methods have transformative potential for parameterized systems and multi-scale models where first principle derivations have remained intractable, such as neuroscience, climate modeling, epidemiology, and the electrical grid.

IV. MATERIALS AND METHODS

The innovations presented here are built upon three key ideas: (i) Overcomplete libraries of candidate functions to represent the dynamics, (ii) sparse regression for selecting a small number of terms and (iii) parsimonious selection of the governing equations via Pareto analysis. Enabling the architecture is the ability to accurately compute derivatives even when noise is present in the measurements. Each method is detailed in what follows.

Building Libraries of Candidate Terms

The sparse regression and discovery method (See Fig. 1) begins by first collecting all the spatial, time series data into a single column vector $\mathbf{U} \in \mathbb{C}^{mn}$ representing data collected over m time points and n spatial locations. We also consider any additional input such as a known potential for the Schrödinger equation, or the magnitude of complex data, in a column vector $\mathbf{Q} \in \mathbb{C}^{mn}$. Next, a library $\mathbf{\Theta}(\mathbf{U}, \mathbf{Q}) \in \mathbb{C}^{mn \times D}$ of D candidate linear and nonlinear terms and partial derivatives for the PDE is constructed. Derivatives are taken either using finite differences for clean data, or when noise is added, with polynomial interpolation. The candidate linear and nonlinear terms and partial derivatives are then combined into a matrix $\mathbf{\Theta}(\mathbf{U}, \mathbf{Q})$ which takes the form, for example:

$$\mathbf{\Theta}(\mathbf{U}, \mathbf{Q}) = \begin{bmatrix} 1 & \mathbf{U} & \mathbf{U}^2 & \ldots & \mathbf{Q} & \ldots & \mathbf{U}_x & \mathbf{U}\mathbf{U}_x & \ldots \end{bmatrix}. \quad (4)$$

Each column of $\mathbf{\Theta}$ contains all of the values of a particular candidate function across all of the mn space-time grid points on which data is collected. Therefore, if we have data on an $n \times m$ grid (e.g. a 256×100 grid represents 256 spatial measurements at 100 time points) and have 50 candidate terms in the PDE, then $\mathbf{\Theta} \in \mathbb{C}^{256 \cdot 100 \times 50}$. The time derivative $\mathbf{U}_t$ is also computed and reshaped into a column vector. Figure 1 demonstrates the data collection and processing. As an example, a column of $\mathbf{\Theta}(\mathbf{U}, \mathbf{Q})$ may be $q^2 u_{xx}$.

Algorithm 1: STRidge($\mathbf{\Theta}, \mathbf{U}_t, \lambda, tol$, iters)

```
$\hat{\xi} = arg\,min_\xi \|\mathbf{\Theta}\xi - \mathbf{U}_t\|_2^2 + \lambda\|\xi\|_2^2$        # ridge regression
bigcoeffs = $\{j : |\hat{\xi}_j| \geq tol\}$        # select large coefficients
$\hat{\xi}[\sim$ bigcoeffs$] = 0$                  # apply hard threshold
$\hat{\xi}$[bigcoeffs] = STRidge($\mathbf{\Theta}$[:, bigcoeffs], $\mathbf{U}_t$, $tol$, iters − 1)
                          # recursive call with fewer coefficients
return $\hat{\xi}$
```

The PDE evolution can be expressed in this library as follows:

$$\mathbf{U}_t = \mathbf{\Theta}(\mathbf{U}, \mathbf{Q})\xi. \quad (5)$$

Each entry in ξ is a coefficient corresponding to a term in the PDE, and for canonical PDEs, the vector ξ is *sparse*, meaning that only a few terms are active.

Note that if we assume $\mathbf{\Theta}$ is an over complete library, meaning $\mathbf{\Theta}$ has a sufficiently rich column space that the dynamics will be in it's range, then the PDE should be well-represented by Eq. (5) with a sparse vector of coefficients ξ. This is equivalent to picking enough candidate functions that the full PDE may be written as a weighted sum of library terms. For an unbiased representation of the dynamics, we would simply solve the least squares problem for ξ. However, even with the only error coming from numerical round-off, the least-squares solution may be inaccurate. In particular, ξ will have predominantly nonzero values suggesting a PDE with every functional form contained in the library. Importantly, regression problems similar to (5) result in least squares problems which are poorly conditioned. Error in computing the derivatives (already an ill-conditioned problem with noise) will be magnified by numerical errors when inverting $\mathbf{\Theta}$. Thus a least squares regression radically changes the qualitative nature of the inferred dynamics.

Sparse Regression

In general, we require the sparsest vector ξ that satisfies (5) with a small residual. Instead of an intractable combinatorial search through all possible sparse vector structures, a common technique is to relax the problem to a convex ℓ_1 regularized least squares [15]; however, this tends to perform poorly with highly correlated data. Instead, we approximate the problem using candidate solutions to a ridge regression problem with hard thresholding, which we call sequential threshold ridge regression (STRidge in Algorithm 1). For a given tolerance and λ, this gives a sparse approximation to ξ. We iteratively refine the tolerance of Algorithm 1 to find the best predictor based on the selection criteria,

$$\hat{\xi} = \text{argmin}_\xi \|\mathbf{\Theta}(\mathbf{U}, \mathbf{Q})\xi - \mathbf{U}_t\|_2^2 + \epsilon\kappa(\mathbf{\Theta}(\mathbf{U}, \mathbf{Q}))\|\xi\|_0 \quad (6)$$

where $\kappa(\mathbf{\Theta})$ is the condition number of the matrix $\mathbf{\Theta}$, indicating stronger regularization for ill-posed problems.

Penalizing $||\xi||_0$ discourages over fitting by selecting from the optimal position in a Pareto front.

Other methods exist for finding sparse solutions to least squares problem. Greedy algorithms have been shown to exhibit good performance on sparse optimization problems including PDE-FIND but in some cases were less reliable than STRidge [20]. While STRidge with normalization works well on almost all of the examples we tested, we do not make the claim that it is optimal. In particular, we did not test the performance of the elastic net algorithm which has been shown to have advantages over LASSO [21]. If additional information regarding the PDE is known, for instance if we know one of the terms is nonzero, then this may be incorporated into the penalty on the coefficients.

Numerical Evaluation of Derivatives

Proper evaluation of the numerical derivatives is the most challenging and critical task for the success of the method. This is particularly true when the discretized solution contains measurement noise. Given the well-known accuracy problems with finite-difference approximations, we experimented with a number of more robust numerical differentiation methods. Smoothing with a Gaussian kernel and Tikhonov differentiation were both investigated, but were difficult to implement due to the difficulty controlling the bias-variance trade off. Spectral differentiation with thresholding for high frequency terms was also considered but not used due to it's restriction to periodic domains and difficulty in implementing an appropriate thesholding function. The most reliable and robust method for computing derivatives from noisy data was polynomial interpolation [22]. For each datapoint where we compute a derivative, a polynomial of degree p was fit to greater than p points and derivatives of the polynomial were taken to approximate those of the numerical data. Points close to the boundaries where it was difficult to fit a polynomial were not used in the regression. This method is far from perfect; data close to the boundaries was difficult to differentiate, and we discovered that this could strongly influence the results of PDE-FIND. For a more principled but involved approach to polynomial differentiation, see [23].

Acknowledgements

J. N. Kutz acknowledges support from the Air Force Office of Scientific Research (FA9550-15-1-0385). S. L. Brunton and J. N. Kutz acknowledge support from the Defense Advanced Research Projects Agency (DARPA contract HR0011-16-C-0016). J.L. Proctor would like to thank Bill and Melinda Gates for their active support of the Institute for Disease Modeling and their sponsorship through the Global Good Fund.

Code and supplementary material:
https://github.com/snagcliffs/PDE-FIND

[1] J. Crutchfield and B. McNamara, Comp. sys. **1**, 417 (1987).

[2] I. Kevrekidis, C. Gear, J. Hyman, P. Kevrekidis, O. Runborg, and C. Theodoropoulos, Comm. Math. Sci. **1**, 715 (2003).

[3] R. Gonzalez-Garcia, R. Rico-Martinez, and I. Kevrekidis, Computers & chemical engineering **22**, S965 (1998).

[4] H. U. Voss, P. Kolodner, M. Abel, and J. Kurths, Physical review letters **83**, 3422 (1999).

[5] G. Sugihara, R. May, H. Ye, C.-h. Hsieh, E. Deyle, M. Fogarty, and S. Munch, Science **338**, 496 (2012).

[6] H. Ye, R. J. Beamish, S. M. Glaser, S. C. Grant, C.-h. Hsieh, L. J. Richards, J. T. Schnute, and G. Sugihara, PNAS **112**, E1569 (2015).

[7] A. J. Majda, C. Franzke, and D. Crommelin, Proceedings of the National Academy of Sciences **106**, 3649 (2009).

[8] D. Giannakis and A. J. Majda, Proceedings of the National Academy of Sciences **109**, 2222 (2012).

[9] A. J. Roberts, *Model emergent dynamics in complex systems* (SIAM, 2014).

[10] M. Schmidt, R. Vallabhajosyula, J. Jenkins, J. Hood, A. Soni, J. Wikswo, and H. Lipson, Phys. bio. **8**, 055011 (2011).

[11] B. Daniels and I. Nemenman, Nat. comm. **6** (2015).

[12] B. Daniels and I. Nemenman, PloS one **10**, e0119821 (2015).

[13] J. Bongard and H. Lipson, PNAS **104**, 9943 (2007).

[14] M. Schmidt and H. Lipson, Science **324**, 81 (2009).

[15] R. Tibshirani, J. Roy. Stat. Soc. B p. 267 (1996).

[16] S. L. Brunton, J. L. Proctor, and J. N. Kutz, PNAS **113**, 3932 (2016).

[17] N. M. Mangan, S. L. Brunton, J. L. Proctor, and J. N. Kutz, ArXiv e-prints arXiv:1605.08368 (2016).

[18] P. Holmes, J. Lumley, G. Berkooz, and C. Rowley, *Turbulence, coherent structures, dynamical systems and symmetry* (Cambridge, Cambridge, England, 2012), 2nd ed.

[19] A. Einstein, Ann. der Physik **17**, 549 (1905).

[20] T. Zhang, in *Advances in Neural Information Processing Systems 21*, edited by D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou (Curran Associates, Inc., 2009), pp. 1921–1928.

[21] H. Zou and T. Hastie, Journal of the Royal Statistical Society: Series B (Statistical Methodology) **67**, 301 (2005).

[22] I. Knowles and R. J. Renka, Electronic Journal of Differential Equations pp. 235–246 (2014).

[23] O. Bruno and D. Hoch, SIAM Journal on Numerical Analysis **50**, 1581 (2012).