CrossMark
click for updates

Discovering governing equations from data by sparse identification of nonlinear dynamical systems

Steven L. Brunton[a,1], Joshua L. Proctor[b], and J. Nathan Kutz[c]

[a]Department of Mechanical Engineering, University of Washington, Seattle, WA 98195; [b]Institute for Disease Modeling, Bellevue, WA 98005; and [c]Department of Applied Mathematics, University of Washington, Seattle, WA 98195

Edited by William Bialek, Princeton University, Princeton, NJ, and approved March 1, 2016 (received for review August 31, 2015)

Extracting governing equations from data is a central challenge in many diverse areas of science and engineering. Data are abundant whereas models often remain elusive, as in climate science, neuroscience, ecology, finance, and epidemiology, to name only a few examples. In this work, we combine sparsity-promoting techniques and machine learning with nonlinear dynamical systems to discover governing equations from noisy measurement data. The only assumption about the structure of the model is that there are only a few important terms that govern the dynamics, so that the equations are sparse in the space of possible functions; this assumption holds for many physical systems in an appropriate basis. In particular, we use sparse regression to determine the fewest terms in the dynamic governing equations required to accurately represent the data. This results in parsimonious models that balance accuracy with model complexity to avoid overfitting. We demonstrate the algorithm on a wide range of problems, from simple canonical systems, including linear and nonlinear oscillators and the chaotic Lorenz system, to the fluid vortex shedding behind an obstacle. The fluid example illustrates the ability of this method to discover the underlying dynamics of a system that took experts in the community nearly 30 years to resolve. We also show that this method generalizes to parameterized systems and systems that are time-varying or have external forcing.

dynamical systems | machine learning | sparse regression | system identification | optimization

Advances in machine learning (1) and data science (2) have promised a renaissance in the analysis and understanding of complex data, extracting patterns in vast multimodal data that are beyond the ability of humans to grasp. However, despite the rapid development of tools to understand static data based on statistical relationships, there has been slow progress in distilling physical models of dynamic processes from big data. This has limited the ability of data science models to extrapolate the dynamics beyond the attractor where they were sampled and constructed.

An analogy may be drawn with the discoveries of Kepler and Newton. Kepler, equipped with the most extensive and accurate planetary data of the era, developed a data-driven model for planetary motion, resulting in his famous elliptic orbits. However, this was an attractor-based view of the world, and it did not explain the fundamental dynamic relationships that give rise to planetary orbits, or provide a model for how these bodies react when perturbed. Newton, in contrast, discovered a dynamic relationship between momentum and energy that described the underlying processes responsible for these elliptic orbits. This dynamic model may be generalized to predict behavior in regimes where no data were collected. Newton's model has proven remarkably robust for engineering design, making it possible to land a spacecraft on the moon, which would not have been possible using Kepler's model alone.

A seminal breakthrough by Bongard and Lipson (3) and Schmidt and Lipson (4) has resulted in a new approach to determine the underlying structure of a nonlinear dynamical system from data. This method uses symbolic regression [i.e., genetic programming (5)] to find nonlinear differential equations, and it balances complexity of the model, measured in the number of terms, with model accuracy. The resulting model identification realizes a long-sought goal of the physics and engineering communities to discover dynamical systems from data. However, symbolic regression is expensive, does not scale well to large systems of interest, and may be prone to overfitting unless care is taken to explicitly balance model complexity with predictive power. In ref. 4, the Pareto front is used to find parsimonious models. There are other techniques that address various aspects of the dynamical system discovery problem. These include methods to discover governing equations from time-series data (6), equation-free modeling (7), empirical dynamic modeling (8, 9), modeling emergent behavior (10), and automated inference of dynamics (11–13); ref. 12 provides an excellent review.

Sparse Identification of Nonlinear Dynamics (SINDy)

In this work, we reenvision the dynamical system discovery problem from the perspective of sparse regression (14–16) and compressed sensing (17–22). In particular, we leverage the fact that most physical systems have only a few relevant terms that define the dynamics, making the governing equations sparse in a high-dimensional nonlinear function space. The combination of sparsity methods in dynamical systems is quite recent (23–30). Here, we consider dynamical systems (31) of the form

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t)). \quad [1]$$

The vector $\mathbf{x}(t) \in \mathbb{R}^n$ denotes the state of a system at time t, and the function $\mathbf{f}(\mathbf{x}(t))$ represents the dynamic constraints that define the equations of motion of the system, such as Newton's second law. Later, the dynamics will be generalized to include parameterization, time dependence, and forcing.

Significance

Understanding dynamic constraints and balances in nature has facilitated rapid development of knowledge and enabled technology, including aircraft, combustion engines, satellites, and electrical power. This work develops a novel framework to discover governing equations underlying a dynamical system simply from data measurements, leveraging advances in sparsity techniques and machine learning. The resulting models are parsimonious, balancing model complexity with descriptive ability while avoiding overfitting. There are many critical data-driven problems, such as understanding cognition from neural recordings, inferring climate patterns, determining stability of financial markets, predicting and suppressing the spread of disease, and controlling turbulence for greener transportation and energy. With abundant data and elusive laws, data-driven discovery of dynamics will continue to play an important role in these efforts.

Author contributions: S.L.B., J.L.P., and J.N.K. designed research; S.L.B. performed research; S.L.B., J.L.P., and J.N.K. analyzed data; and S.L.B. wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

Freely available online through the PNAS open access option.

[1]To whom correspondence should be addressed. Email: sbrunton@uw.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1517384113/-/DCSupplemental.

Downloaded from https://www.pnas.org by 46.12.210.136 on November 19, 2026 from IP address 46.12.210.136.

The key observation is that for many systems of interest, the function $\mathbf{f}$ consists of only a few terms, making it sparse in the space of possible functions. Recent advances in compressed sensing and sparse regression make this viewpoint of sparsity favorable, because it is now possible to determine which right-hand-side terms are nonzero without performing a combinatorially intractable brute-force search. This guarantees that the sparse solution is found with high probability using convex methods that scale to large problems favorably with Moore's law. The resulting sparse model identification inherently balances model complexity (i.e., sparsity of the right-hand-side dynamics) with accuracy, avoiding overfitting the model to the data. Wang et al. (23) have used compressed sensing to identify nonlinear dynamics and predict catastrophes; here, we advocate using sparse regression to mitigate noise.

To determine the function $\mathbf{f}$ from data, we collect a time history of the state $\mathbf{x}(t)$ and either measure the derivative $\dot{\mathbf{x}}(t)$ or approximate it numerically from $\mathbf{x}(t)$. The data are sampled at several times $t_1, t_2, \cdots, t_m$ and arranged into two matrices:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \mathbf{x}^T(t_2) \\ \vdots \\ \mathbf{x}^T(t_m) \end{bmatrix} = \overbrace{\begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m) \end{bmatrix}}^{\text{state}} \downarrow \text{time}$$

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{\mathbf{x}}^T(t_1) \\ \dot{\mathbf{x}}^T(t_2) \\ \vdots \\ \dot{\mathbf{x}}^T(t_m) \end{bmatrix} = \begin{bmatrix} \dot{x}_1(t_1) & \dot{x}_2(t_1) & \cdots & \dot{x}_n(t_1) \\ \dot{x}_1(t_2) & \dot{x}_2(t_2) & \cdots & \dot{x}_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \dot{x}_1(t_m) & \dot{x}_2(t_m) & \cdots & \dot{x}_n(t_m) \end{bmatrix}.$$

Next, we construct a library $\Theta(\mathbf{X})$ consisting of candidate nonlinear functions of the columns of $\mathbf{X}$. For example, $\Theta(\mathbf{X})$ may consist of constant, polynomial, and trigonometric terms:

$$\Theta(\mathbf{X}) = \begin{bmatrix} | & | & | & | & & | & | & \\ 1 & \mathbf{X} & \mathbf{X}^{P_2} & \mathbf{X}^{P_3} & \cdots & \sin(\mathbf{X}) & \cos(\mathbf{X}) & \cdots \\ | & | & | & | & & | & | & \end{bmatrix}. \quad [2]$$

Here, higher polynomials are denoted as $\mathbf{X}^{P_2}, \mathbf{X}^{P_3}$, etc., where $\mathbf{X}^{P_2}$ denotes the quadratic nonlinearities in the state $\mathbf{x}$:

$$\mathbf{X}^{P_2} = \begin{bmatrix} x_1^2(t_1) & x_1(t_1)x_2(t_1) & \cdots & x_2^2(t_1) & \cdots & x_n^2(t_1) \\ x_1^2(t_2) & x_1(t_2)x_2(t_2) & \cdots & x_2^2(t_2) & \cdots & x_n^2(t_2) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x_1^2(t_m) & x_1(t_m)x_2(t_m) & \cdots & x_2^2(t_m) & \cdots & x_n^2(t_m) \end{bmatrix}.$$

Each column of $\Theta(\mathbf{X})$ represents a candidate function for the right-hand side of Eq. **1**. There is tremendous freedom in choosing the entries in this matrix of nonlinearities. Because we believe that only a few of these nonlinearities are active in each row of $\mathbf{f}$, we may set up a sparse regression problem to determine the sparse vectors of coefficients $\Xi = [\boldsymbol{\xi}_1\ \boldsymbol{\xi}_2\ \cdots\ \boldsymbol{\xi}_n]$ that determine which nonlinearities are active:

$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi. \quad [3]$$

This is illustrated in Fig. 1. Each column $\boldsymbol{\xi}_k$ of Ξ is a sparse vector of coefficients determining which terms are active in the right-hand side for one of the row equations $\dot{\mathbf{x}}_k = \mathbf{f}_k(\mathbf{x})$ in Eq. **1**. Once Ξ has been determined, a model of each row of the governing equations may be constructed as follows:

$$\dot{\mathbf{x}}_k = \mathbf{f}_k(\mathbf{x}) = \Theta(\mathbf{x}^T)\boldsymbol{\xi}_k. \quad [4]$$

Note that $\Theta(\mathbf{x}^T)$ is a vector of symbolic functions of elements of $\mathbf{x}$, as opposed to $\Theta(\mathbf{X})$, which is a data matrix. Thus,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) = \Xi^T\left(\Theta(\mathbf{x}^T)\right)^T. \quad [5]$$

Each column of Eq. **3** requires a distinct optimization to find the sparse vector of coefficients $\boldsymbol{\xi}_k$ for the kth row equation. We may also normalize the columns of $\Theta(\mathbf{X})$, especially when entries of $\mathbf{X}$ are small, as discussed in the *SI Appendix*.

For examples in this paper, the matrix $\Theta(\mathbf{X})$ has size $m \times p$, where p is the number of candidate functions, and $m \gg p$ because there are more data samples than functions; this is possible in a restricted basis, such as the polynomial basis in Eq. **2**. In practice, it may be helpful to test many different function bases and use the sparsity and accuracy of the resulting model as a diagnostic tool to determine the correct basis to represent the dynamics in. In *SI Appendix*, Appendix B, two examples are explored where the sparse identification algorithm fails because the dynamics are not sparse in the chosen basis.

Realistically, often only $\mathbf{X}$ is available, and $\dot{\mathbf{X}}$ must be approximated numerically, as in all of the continuous-time examples below. Thus, $\mathbf{X}$ and $\dot{\mathbf{X}}$ are contaminated with noise so Eq. **3** does not hold exactly. Instead,

$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi + \eta\mathbf{Z}, \quad [6]$$

where $\mathbf{Z}$ is modeled as a matrix of independent identically distributed Gaussian entries with zero mean, and noise magnitude η. Thus, we seek a sparse solution to an overdetermined system with noise. The least absolute shrinkage and selection operator (LASSO) (14, 15) is an ℓ_1-regularized regression that promotes sparsity and works well with this type of data. However, it may be computationally expensive for very large data sets. An alternative based on sequential thresholded least-squares is presented in Code 1 in the *SI Appendix*.

Depending on the noise, it may be necessary to filter $\mathbf{X}$ and $\dot{\mathbf{X}}$ before solving for Ξ. In many of the examples below, only the data $\mathbf{X}$ are available, and $\dot{\mathbf{X}}$ are obtained by differentiation. To counteract differentiation error, we use the total variation regularization (32) to denoise the derivative (33). This works quite well when only state data $\mathbf{X}$ are available, as illustrated on the Lorenz system (*SI Appendix*, Fig. S7). Alternatively, the data $\mathbf{X}$ and $\dot{\mathbf{X}}$ may be filtered, for example using the optimal hard threshold for singular values described in ref. 34. Insensitivity to noise is a critical feature of an algorithm that identifies dynamics from data (11–13).

Often, the physical system of interest may be naturally represented by a partial differential equation (PDE) in a few spatial variables. If data are collected from a numerical discretization or from experimental measurements on a spatial grid, then the state dimension n may be prohibitively large. For example, in fluid dynamics, even simple 2D and 3D flows may require tens of thousands up to billions of variables to represent the discretized system. The proposed method is ill-suited for a large state dimension n, because of the factorial growth of Θ in n and because each of the n row equations in Eq. **4** requires a separate optimization. Fortunately, many high-dimensional systems of interest evolve on a low-dimensional manifold or attractor that is well-approximated using a low-rank basis Ψ (35, 36). For example, if data $\mathbf{X}$ are collected for a high-dimensional system as in Eq. **2**, it is possible to obtain a low-rank approximation using dimensionality reduction techniques, such as the proper orthogonal decomposition (POD) (35, 37).

The proposed sparse identification of nonlinear dynamics (SINDy) method depends on the choice of measurement variables, data quality, and the sparsifying function basis. There is no single method that will solve all problems in nonlinear system identification, but this method highlights the importance of these underlying choices and can help guide the analysis. The challenges of choosing measurement variables and a sparsifying function basis are explored in *SI Appendix*, section 4.5 and Appendixes A and B.

Simply put, we need the right coordinates and function basis to yield sparse dynamics; the feasibility and flexibility of these requirements is discussed in *Discussion* and *SI Appendix* section 4.5

Downloaded from https://www.pnas.org by 46.12.210.136 on November 19, 2026 from IP address 46.12.210.136.

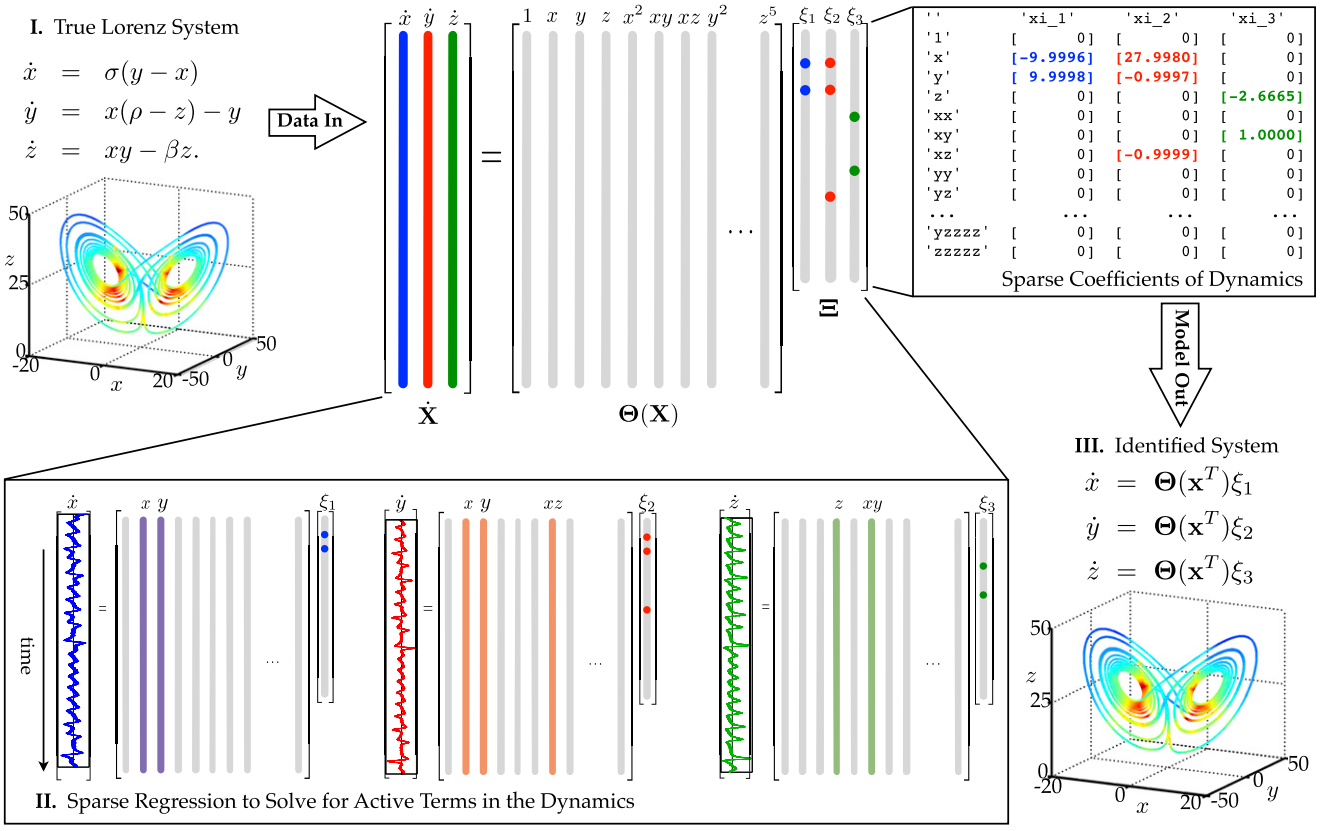

Fig. 1. Schematic of the SINDy algorithm, demonstrated on the Lorenz equations. Data are collected from the system, including a time history of the states $\mathbf{X}$ and derivatives $\dot{\mathbf{X}}$; the assumption of having $\dot{\mathbf{X}}$ is relaxed later. Next, a library of nonlinear functions of the states, $\Theta(\mathbf{X})$, is constructed. This nonlinear feature library is used to find the fewest terms needed to satisfy $\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi$. The few entries in the vectors of Ξ, solved for by sparse regression, denote the relevant terms in the right-hand side of the dynamics. Parameter values are $\sigma = 10, \beta = 8/3, \rho = 28$, $(x_0, y_0, z_0)^T = (-8, 7, 27)^T$. The trajectory on the Lorenz attractor is colored by the adaptive time step required, with red indicating a smaller time step.

and Appendixes A and B. However, it may be difficult to know the correct variables a priori. Fortunately, time-delay coordinates may provide useful variables from a time series (9, 12, 38). The ability to reconstruct sparse attractor dynamics using time-delay coordinates is demonstrated in *SI Appendix*, section 4.5 using a single variable of the Lorenz system.

The choice of coordinates and the sparsifying basis are intimately related, and the best choice is not always clear. However, basic knowledge of the physics (e.g., Navier–Stokes equations have quadratic nonlinearities, and the Schrödinger equation has $|\mathbf{x}|^2\mathbf{x}$ terms) may provide a reasonable choice of nonlinear functions and measurement coordinates. In fact, the sparsity and accuracy of the proposed sparse identified model may provide valuable diagnostic information about the correct measurement coordinates and basis in which to represent the dynamics. Choosing the right coordinates to simplify dynamics has always been important, as exemplified by Lagrangian and Hamiltonian mechanics (39). There is still a need for experts to find and exploit symmetry in the system, and the proposed methods should be complemented by advanced algorithms in machine learning to extract useful features.

Results

We demonstrate the algorithm on canonical systems*, ranging from linear and nonlinear oscillators (*SI Appendix*, section 4.1), to noisy measurements of the chaotic Lorenz system, to the unsteady fluid wake behind a cylinder, extending this method to nonlinear PDEs and high-dimensional data. Finally, we show that bifurcation parameters may be included in the models, recovering the parameterized logistic map and the Hopf normal form from noisy measurements. In each example, we explore the ability to identify the dynamics from state measurements alone, without access to derivatives.

It is important to reiterate that the sparse identification method relies on a fortunate choice of coordinates and function basis that facilitate sparse representation of the dynamics. In *SI Appendix*, Appendix B, we explore the limitations of the method for examples where these assumptions break down: the Lorenz system transformed into nonlinear coordinates and the glycolytic oscillator model (11–13).

Chaotic Lorenz System. As a first example, consider a canonical model for chaotic dynamics, the Lorenz system (40):

$$\dot{x} = \sigma(y - x), \qquad [7a]$$

$$\dot{y} = x(\rho - z) - y, \qquad [7b]$$

$$\dot{z} = xy - \beta z. \qquad [7c]$$

Although these equations give rise to rich and chaotic dynamics that evolve on an attractor, there are only a few terms in the right-hand side of the equations. Fig. 1 shows a schematic of how data are collected for this example, and how sparse dynamics are identified in a space of possible right-hand-side functions using convex ℓ_1 minimization.

For this example, data are collected for the Lorenz system, and stacked into two large data matrices $\mathbf{X}$ and $\dot{\mathbf{X}}$, where each row of $\mathbf{X}$ is a snapshot of the state $\mathbf{x}$ in time, and each row of $\dot{\mathbf{X}}$ is a snapshot

*Code is available at faculty.washington.edu/sbrunton/sparsedynamics.zip.

Downloaded from https://www.pnas.org by 46.12.210.136 on November 19, 2026 from IP address 46.12.210.136.

of the time derivative of the state $\dot{\mathbf{x}}$ in time. Here, the right-hand-side dynamics are identified in the space of polynomials $\Theta(\mathbf{X})$ in (x,y,z) up to fifth order, although other functions such as sin, cos, exp, or higher-order polynomials may be included:

$$\Theta(\mathbf{X}) = \begin{bmatrix} | & | & | & | & | & \cdots & | \\ x(t) & \mathbf{y}(t) & \mathbf{z}(t) & \mathbf{x}(t)^2 & \mathbf{x}(t)\mathbf{y}(t) & \cdots & \mathbf{z}(t)^5 \\ | & | & | & | & | & \cdots & | \end{bmatrix}.$$

Each column of $\Theta(\mathbf{X})$ represents a candidate function for the right-hand side of Eq. **1**. Because only a few of these terms are active in each row of $\mathbf{f}$, we solve the sparse regression problem in Eq. **3** to determine the sparse vectors of coefficients $\Xi = [\boldsymbol{\xi}_1\ \boldsymbol{\xi}_2 \cdots \boldsymbol{\xi}_n]$ that determine which terms are active in the dynamics. This is illustrated schematically in Fig. 1, where sparse vectors $\boldsymbol{\xi}_k$ are found to represent the derivative $\dot{x}_k$ as a linear combination of the fewest terms in $\Theta(\mathbf{X})$.

In the Lorenz example, the ability to capture dynamics on the attractor is more important than the ability to predict an individual trajectory, because chaos will quickly cause any small variations in initial conditions or model coefficients to diverge exponentially. As shown in Fig. 1, the sparse model accurately reproduces the attractor dynamics from chaotic trajectory measurements. The algorithm not only identifies the correct terms in the dynamics, but it accurately determines the coefficients to within .03% of the true values. We also explore the identification of the dynamics when only noisy state measurements are available (*SI Appendix*, Fig. S7). The correct dynamics are identified, and the attractor is preserved for surprisingly large noise values. In *SI Appendix*, section 4.5, we reconstruct the attractor dynamics in the Lorenz system using time-delay coordinates from a single measurement $x(t)$.

PDE for Vortex Shedding Behind an Obstacle. The Lorenz system is a low-dimensional model of more realistic high-dimensional PDE models for fluid convection in the atmosphere. Many systems of interest are governed by PDEs (24), such as weather and climate, epidemiology, and the power grid, to name a few. Each of these examples is characterized by big data, consisting of large spatially resolved measurements consisting of millions or billions of states and spanning orders of magnitude of scale in both space and time. However, many high-dimensional, real-world systems evolve on a low-dimensional attractor, making the effective dimension much smaller (35).

Here we generalize the SINDy method to an example in fluid dynamics that typifies many of the challenges outlined above. In the context of data from a PDE, our algorithm shares some connections to the dynamic mode decomposition, which is a linear dynamic regression (41–43). Data are collected for the fluid flow past a cylinder at Reynolds number 100 using direct numerical simulations of the 2D Navier–Stokes equations (44, 45). The nonlinear dynamic relationship between the dominant coherent structures is identified from these flow-field measurements with no knowledge of the governing equations.

The flow past a cylinder is a particularly interesting example because of its rich history in fluid mechanics and dynamical systems. It has long been theorized that turbulence is the result of a series of Hopf bifurcations that occur as the flow velocity increases (46), giving rise to more rich and intricate structures in the fluid. After 15 years, the first Hopf bifurcation was discovered in a fluid system, in the transition from a steady laminar wake to laminar periodic vortex shedding at Reynolds number 47 (47, 48). This discovery led to a long-standing debate about how a Hopf bifurcation, with cubic nonlinearity, can be exhibited in a Navier–Stokes fluid with quadratic nonlinearities. After 15 more years, this was resolved using a separation of timescales and a mean-field model (49), shown in Eq. **8**. It was shown that coupling between oscillatory modes and the base flow gives rise to a slow manifold (Fig. 2, *Left*), which results in algebraic terms that approximate cubic nonlinearities on slow timescales.

This example provides a compelling test case for the proposed algorithm, because the underlying form of the dynamics took nearly three decades for experts in the community to uncover. Because the state dimension is large, consisting of the fluid state at hundreds of thousands of grid points, it is first necessary to reduce the dimension of the system. The POD (35, 37), provides a low-rank basis resulting in a hierarchy of orthonormal modes that, when truncated, capture the most energy of the original system for the given rank truncation. The first two most energetic POD modes capture a significant portion of the energy, and steady-state vortex shedding is a limit cycle in these coordinates. An additional mode, called the shift mode, is included to capture the transient dynamics connecting the unstable steady state ("C" in Fig. 2) with the mean of the limit cycle (49) ("B" in Fig. 2). These modes define the x,y,z coordinates in Fig. 2.

In the coordinate system described above, the mean-field model for the cylinder dynamics is given by (49)

$$\dot{x} = \mu x - \omega y + Axz, \qquad \textbf{[8a]}$$

$$\dot{y} = \omega x + \mu y + Ayz, \qquad \textbf{[8b]}$$

$$\dot{z} = -\lambda\left(z - x^2 - y^2\right). \qquad \textbf{[8c]}$$

If λ is large, so that the z dynamics are fast, then the mean flow rapidly corrects to be on the (slow) manifold $z = x^2 + y^2$ given by

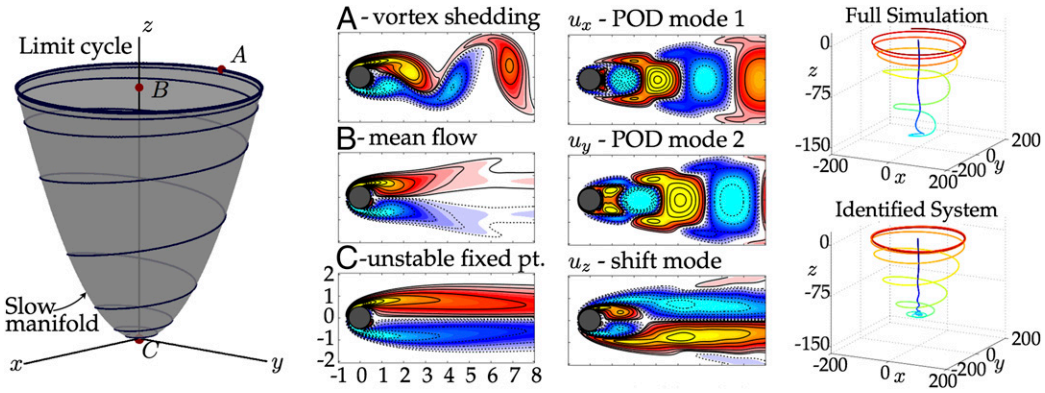

Fig. 2. Example of high-dimensional dynamical system from fluid dynamics. The vortex shedding past a cylinder is a prototypical example that is used for flow control, with relevance to many applications, including drag reduction behind vehicles. The vortex shedding is the result of a Hopf bifurcation. However, because the Navier–Stokes equations have quadratic nonlinearity, it is necessary to use a mean-field model with a separation of timescales, where a fast mean-field deformation is slave to the slow vortex shedding dynamics. The parabolic slow manifold is shown (*Left*), with the unstable fixed point (*C*), mean flow (*B*), and vortex shedding (*A*). A POD basis and shift mode are used to reduce the dimension of the problem (*Middle Right*). The identified dynamics closely match the true trajectory in POD coordinates, and most importantly, they capture the quadratic nonlinearity and timescales associated with the mean-field model.

Downloaded from https://www.pnas.org by 46.12.210.136 on November 19, 2026 from IP address 46.12.210.136.

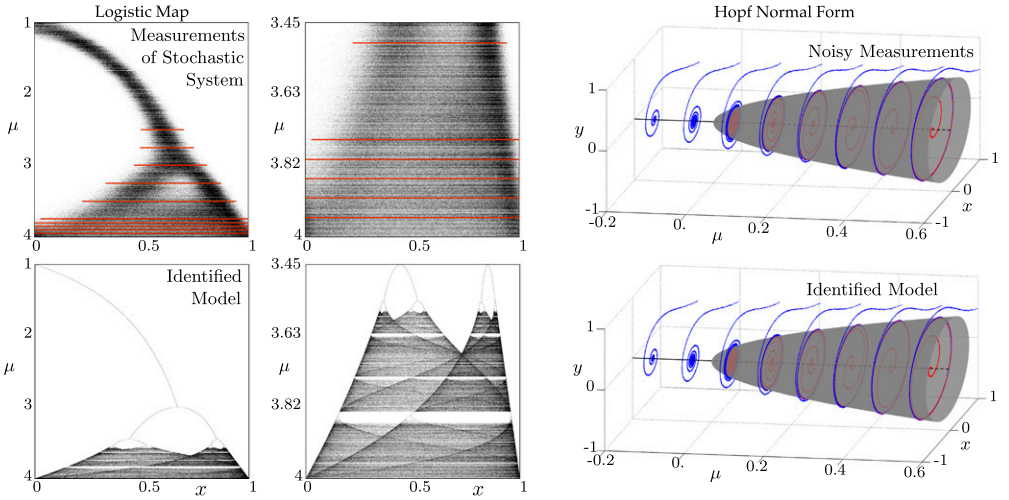

Fig. 3. SINDy algorithm is able to identify normal forms and capture bifurcations, as demonstrated on the logistic map (*Left*) and the Hopf normal form (*Right*). Noisy data from both systems are used to train models. For the logistic map, a handful of parameter values μ (red lines), are used for the training data, and the correct normal form and bifurcation sequence is captured (below). Noisy data for the Hopf normal form are collected at a few values of μ, and the total variation derivative (33) is used to compute time derivatives. The accurate Hopf normal form is reproduced (below). The nonlinear terms identified by the algorithm are in *SI Appendix*, section 4.4 and Appendix C.

the amplitude of vortex shedding. When substituting this algebraic relationship into Eqs. **8a** and **8b**, we recover the Hopf normal form on the slow manifold.

With a time history of these three coordinates, the proposed algorithm correctly identifies quadratic nonlinearities and reproduces a parabolic slow manifold. Note that derivative measurements are not available, but are computed from the state variables. Interestingly, when the training data do not include trajectories that originate off of the slow manifold, the algorithm incorrectly identifies cubic nonlinearities, and fails to identify the slow manifold.

Normal Forms, Bifurcations, and Parameterized Systems. In practice, many real-world systems depend on parameters, and dramatic changes, or bifurcations, may occur when the parameter is varied. The SINDy algorithm is readily extended to encompass these important parameterized systems, allowing for the discovery of normal forms (31, 50) associated with a bifurcation parameter μ. First, we append μ to the dynamics:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x};\mu), \qquad [9a]$$

$$\dot{\mu} = 0. \qquad [9b]$$

It is then possible to identify $\mathbf{f}(\mathbf{x};\mu)$ as a sparse combination of functions of $\mathbf{x}$ as well as the bifurcation parameter μ.

Identifying parameterized dynamics is shown in two examples: the 1D logistic map with stochastic forcing,

$$x_{k+1} = \mu x_k(1-x_k) + \eta_k,$$

and the 2D Hopf normal form (51),

$$\dot{x} = \mu x + \omega y - A x\left(x^2+y^2\right)$$

$$\dot{y} = -\omega x + \mu y - A y\left(x^2+y^2\right).$$

The logistic map is a classical model for population dynamics, and the Hopf normal form models spontaneous oscillations in chemical reactions, electrical circuits, and fluid instability.

The noisy measurements and the sparse dynamic reconstructions for both examples are shown in Fig. 3. In the logistic map example, the stochastically forced trajectory is sampled at 10 discrete parameter values, shown in red. From these measurements, the correct parameterized dynamics are identified. The parameterization is accurate enough to capture the cascade of bifurcations as μ is increased, resulting in the detailed bifurcation diagram in Fig. 3. Parameters are identified to within .1% of true values (*SI Appendix*, Appendix C).

In the Hopf normal-form example, noisy state measurements from eight parameter values are sampled, with data collected on the blue and red trajectories in Fig. 3 (*Top Right*). Noise is added to the position measurements to simulate sensor noise, and the total variation regularized derivative (33) is used. In this example, the normal form is correctly identified, resulting in accurate limit cycle amplitudes and growth rates (*Bottom Right*). The correct identification of a normal form depends critically on the choice of variables and the nonlinear basis functions used for $\Theta(\mathbf{x})$. In practice, these choices may be informed by machine learning and data mining, by partial knowledge of the physics, and by expert human intuition.

Similarly, time dependence and external forcing or feedback control $\mathbf{u}(t)$ may be added to the vector field:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}(t), t),$$

$$\dot{t} = 1.$$

Generalizing the SINDy algorithm makes it possible to analyze systems that are externally forced or controlled. For example, the climate is both parameterized (50) and has external forcing, including carbon dioxide and solar radiation. The financial market is another important example with forcing and active feedback control.

Discussion

In summary, we have demonstrated a powerful technique to identify nonlinear dynamical systems from data without assumptions on the form of the governing equations. This builds on prior work in symbolic regression but with innovations related to sparse regression, which allow our algorithms to scale to high-dimensional systems. We demonstrate this method on a number of example systems exhibiting chaos, high-dimensional data with low-rank coherence, and parameterized dynamics. As shown in the Lorenz example, the ability to predict a specific trajectory may be less important than the ability to capture the attractor dynamics. The example from fluid dynamics highlights the remarkable ability of this method to extract dynamics in a fluid system that took three decades for experts in the community to explain. There are numerous fields where this method may be applied, where there are ample data and the absence of governing equations, including neuroscience, climate science, epidemiology, and financial markets. Finally, normal forms may be discovered by including parameters in the optimization, as shown in two examples. The identification of sparse governing equations and parameterizations marks a significant step toward the long-held goal of intelligent, unassisted identification of dynamical systems.

We have demonstrated the robustness of the sparse dynamics algorithm to measurement noise and unavailability of derivative

Downloaded from https://www.pnas.org by 46.12.210.136 on November 19, 2026 from IP address 46.12.210.136.

measurements in the Lorenz system (*SI Appendix*, Figs. S6 and S7), logistic map (*SI Appendix*, section 4.4.1), and Hopf normal form (*SI Appendix*, section 4.4.2) examples. In each case, the sparse regression framework appears well-suited to measurement and process noise, especially when derivatives are smoothed using the total-variation regularized derivative.

A significant outstanding issue in the above approach is the correct choice of measurement coordinates and the choice of sparsifying function basis for the dynamics. As shown in *SI Appendix*, Appendix B, the algorithm fails to identify an accurate sparse model when the measurement coordinates and function basis are not amenable to sparse representation. In the successful examples, the coordinates and function spaces were somehow fortunate in that they enabled sparse representation. There is no simple solution to this challenge, and there must be a coordinated effort to incorporate expert knowledge, feature extraction, and other advanced methods. However, in practice, there may be some hope of obtaining the correct coordinate system and function basis without knowing the solution ahead of time, because we often know something about the physics that guide the choice of function space. The failure to identify an accurate sparse model also provides valuable diagnostic information about the coordinates and basis. If we have few measurements, these may be augmented using time-delay coordinates, as demonstrated on the Lorenz system (*SI Appendix*, section 4.5). When there are too many measurements, we may extract coherent structures using dimensionality reduction. We also demonstrate the use of polynomial bases to approximate Taylor series of nonlinear dynamics (*SI Appendix*, Appendix A). The connection between sparse optimization and dynamical systems will hopefully spur developments to automate and improve these choices.

Data science is not a panacea for all problems in science and engineering, but used in the right way, it provides a principled approach to maximally leverage the data that we have and inform what new data to collect. Big data are happening all across the sciences, where the data are inherently dynamic, and where traditional approaches are prone to overfitting. Data discovery algorithms that produce parsimonious models are both rare and desirable. Data science will only become more critical to efforts in science in engineering, such as understanding the neural basis of cognition, extracting and predicting coherent changes in the climate, stabilizing financial markets, managing the spread of disease, and controlling turbulence, where data are abundant, but physical laws remain elusive.

ACKNOWLEDGMENTS. We are grateful for discussions with Bing Brunton and Bernd Noack, and for insightful comments from the referees. We also thank Tony Roberts and Jean-Christophe Loiseau. S.L.B. acknowledges support from the University of Washington. J.L.P. thanks Bill and Melinda Gates for their support of the Institute for Disease Modeling and their sponsorship through the Global Good Fund. J.N.K. acknowledges support from the US Air Force Office of Scientific Research (FA9550-09-0174).

1. Jordan MI, Mitchell TM (2015) Machine learning: Trends, perspectives, and prospects. *Science* 349(6245):255–260.
2. Marx V (2013) Biology: The big challenges of big data. *Nature* 498(7453):255–260.
3. Bongard J, Lipson H (2007) Automated reverse engineering of nonlinear dynamical systems. *Proc Natl Acad Sci USA* 104(24):9943–9948.
4. Schmidt M, Lipson H (2009) Distilling free-form natural laws from experimental data. *Science* 324(5923):81–85.
5. Koza JR (1992) *Genetic Programming: On the Programming of Computers by Means of Natural Selection* (MIT Press, Cambridge, MA), Vol 1.
6. Crutchfield JP, McNamara BS (1987) Equations of motion from a data series. *Complex Syst* 1(3):417–452.
7. Kevrekidis IG, et al. (2003) Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis. *Commun Math Sci* 1(4):715–762.
8. Sugihara G, et al. (2012) Detecting causality in complex ecosystems. *Science* 338(6106):496–500.
9. Ye H, et al. (2015) Equation-free mechanistic ecosystem forecasting using empirical dynamic modeling. *Proc Natl Acad Sci USA* 112(13):E1569–E1576.
10. Roberts AJ (2014) *Model Emergent Dynamics in Complex Systems* (SIAM, Philadelphia).
11. Schmidt MD, et al. (2011) Automated refinement and inference of analytical models for metabolic networks. *Phys Biol* 8(5):055011.
12. Daniels BC, Nemenman I (2015) Automated adaptive inference of phenomenological dynamical models. *Nat Commun* 6:8133.
13. Daniels BC, Nemenman I (2015) Efficient inference of parsimonious phenomenological models of cellular dynamics using S-systems and alternating regression. *PLoS One* 10(3):e0119821.
14. Tibshirani R (1996) Regression shrinkage and selection via the lasso. *J R Stat Soc, B* 58(1):267–288.
15. Hastie T, Tibshirani R, Friedman J (2009) *The Elements of Statistical Learning* (Springer, New York), Vol 2.
16. James G, Witten D, Hastie T, Tibshirani R (2013) *An Introduction to Statistical Learning* (Springer, New York).
17. Donoho DL (2006) Compressed sensing. *IEEE Trans Inf Theory* 52(4):1289–1306.
18. Candès EJ, Romberg J, Tao T (2006) Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans Inf Theory* 52(2):489–509.
19. Candès EJ, Romberg J, Tao T (2006) Stable signal recovery from incomplete and inaccurate measurements. *Commun Pure Appl Math* 59(8):1207–1223.
20. Candès EJ, Wakin MB (2008) An introduction to compressive sampling. *IEEE Signal Processing Magazine* 25(2):21–30.
21. Baraniuk RG (2007) Compressive sensing. *IEEE Signal Process Mag* 24(4):118–120.
22. Tropp JA, Gilbert AC (2007) Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans Inf Theory* 53(12):4655–4666.
23. Wang WX, Yang R, Lai YC, Kovanis V, Grebogi C (2011) Predicting catastrophes in nonlinear dynamical systems by compressive sensing. *Phys Rev Lett* 106(15):154101.
24. Schaeffer H, Caflisch R, Hauck CD, Osher S (2013) Sparse dynamics for partial differential equations. *Proc Natl Acad Sci USA* 110(17):6634–6639.
25. Ozoliņš V, Lai R, Caflisch R, Osher S (2013) Compressed modes for variational problems in mathematics and physics. *Proc Natl Acad Sci USA* 110(46):18368–18373.
26. Mackey A, Schaeffer H, Osher S (2014) On the compressive spectral method. *Multiscale Model Simul* 12(4):1800–1827.
27. Brunton SL, Tu JH, Bright I, Kutz JN (2014) Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems. *SIAM J Appl Dyn Syst* 13(4):1716–1732.
28. Proctor JL, Brunton SL, Brunton BW, Kutz JN (2014) Exploiting sparsity and equation-free architectures in complex systems (invited review). *Eur Phys J Spec Top* 223:2665–2684.
29. Bai Z, et al. (2014) Low-dimensional approach for reconstruction of airfoil data via compressive sensing. *AIAA J* 53(4):920–930.
30. Arnaldo I, O'Reilly UM, Veeramachaneni K (2015) Building predictive models via feature synthesis. Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation (ACM, New York), pp 983–990.
31. Holmes P, Guckenheimer J (1983) Nonlinear oscillations, dynamical systems, and bifurcations of vector fields. *Applied Mathematical Sciences* (Springer, Berlin), Vol 42.
32. Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. *Physica D* 60(1):259–268.
33. Chartrand R (2011) Numerical differentiation of noisy, nonsmooth data. *ISRN Applied Mathematics* 2011(2011):164564.
34. Gavish M, Donoho DL (2014) The optimal hard threshold for singular values is $4/\sqrt{3}$. *IEEE Trans Inf Theory* 60(8):5040–5053.
35. Holmes PJ, Lumley JL, Berkooz G, Rowley CW (2012) *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge Monographs in Mechanics (Cambridge Univ Press, Cambridge, UK), 2nd Ed.
36. Majda AJ, Harlim J (2007) Information flow between subspaces of complex dynamical systems. *Proc Natl Acad Sci USA* 104(23):9558–9563.
37. Berkooz G, Holmes P, Lumley JL (1993) The proper orthogonal decomposition in the analysis of turbulent flows. *Annu Rev Fluid Mech* 23:539–575.
38. Takens F (1981) Detecting strange attractors in turbulence. *Lect Notes Math* 898:366–381.
39. Marsden JE, Ratiu TS (1999) *Introduction to Mechanics and Symmetry* (Springer, New York), 2nd Ed.
40. Lorenz EN (1963) Deterministic nonperiodic flow. *J Atmos Sci* 20:130–141.
41. Rowley CW, Mezić I, Bagheri S, Schlatter P, Henningson D (2009) Spectral analysis of nonlinear flows. *J Fluid Mech* 645:115–127.
42. Schmid PJ (2010) Dynamic mode decomposition of numerical and experimental data. *J Fluid Mech* 656:5–28.
43. Mezic I (2013) Analysis of fluid flows via spectral properties of the Koopman operator. *Annu Rev Fluid Mech* 45:357–378.
44. Taira K, Colonius T (2007) The immersed boundary method: A projection approach. *J Comput Phys* 225(2):2118–2137.
45. Colonius T, Taira K (2008) A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. *Comput Methods Appl Mech Eng* 197:2131–2146.
46. Ruelle D, Takens F (1971) On the nature of turbulence. *Commun Math Phys* 20:167–192.
47. Jackson CP (1987) A finite-element study of the onset of vortex shedding in flow past variously shaped bodies. *J Fluid Mech* 182:23–45.
48. Zebib Z (1987) Stability of viscous flow past a circular cylinder. *J Eng Math* 21:155–165.
49. Noack BR, Afanasiev K, Morzynski M, Tadmor G, Thiele F (2003) A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J Fluid Mech* 497:335–363.
50. Majda AJ, Franzke C, Crommelin D (2009) Normal forms for reduced stochastic climate models. *Proc Natl Acad Sci USA* 106(10):3649–3653.
51. Marsden JE, McCracken M (1976) *The Hopf Bifurcation and Its Applications* (Springer, New York), Vol 19.

Downloaded from https://www.pnas.org by 46.12.210.136 on November 19, 2026 from IP address 46.12.210.136.