Reconstruction of normal forms by learning informed observation geometries from data

Or Yair[a], Ronen Talmon[a,1], Ronald R. Coifman[b,1], and Ioannis G. Kevrekidis[c,d,1]

[a]Viterbi Faculty of Electrical Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel; [b]Department of Mathematics, Yale University, New Haven, CT 06511; [c]Department of Chemical and Biological Engineering, Princeton University, Princeton, NJ 08544; and [d]Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544

Contributed by Ronald R. Coifman, July 10, 2017 (sent for review December 7, 2016; reviewed by Alexandre J. Chorin and Guillermo Sapiro)

The discovery of physical laws consistent with empirical observations is at the heart of (applied) science and engineering. These laws typically take the form of nonlinear differential equations depending on parameters; dynamical systems theory provides, through the appropriate normal forms, an "intrinsic" prototypical characterization of the types of dynamical regimes accessible to a given model. Using an implementation of data-informed geometry learning, we directly reconstruct the relevant "normal forms": a quantitative mapping from empirical observations to prototypical realizations of the underlying dynamics. Interestingly, the state variables and the parameters of these realizations are inferred from the empirical observations; without prior knowledge or understanding, they parametrize the dynamics intrinsically without explicit reference to fundamental physical quantities.

dynamical systems | geometry | graph theory | data analysis | empirical models

Introduction

Consider the four sketches displayed in row 1 of Fig. 1. Solely by observation, one could recognize that each image is a phase portrait in a sequence involving a Hopf bifurcation. Taking into account the four images together, one might even identify the trend—that this sequence should be associated with a supercritical Hopf bifurcation, where a stable limit cycle is "born" somewhere between Fig. 1 *B* and *C* as the steady state loses stability at some critical parameter value. It is "obvious" from the transients that the steady state becomes less attracting between Fig. 1 *A* and *B*; the amplitude of the limit cycle gradually grows away from the critical parameter value, and its period should be related to the rate of spiraling of the transients around the steady states.

If we now erased the labels in Fig. 1 and shuffled the four cards, one could easily figure out how to put them back in the "right" order (in effect understanding that this is "naturally" a one-parameter family of images). By recalling the normal form of a Hopf bifurcation, one could attempt to fit some coefficients, find the period and its rate of change with the parameter, and arguably, even estimate where to quantitatively pin Fig. 1*C* in a parametric interval with edges that are Fig. 1 *B* and *D*.

This caricature shows our ability to recognize dynamic patterns, to focus on the salient features, and to make predictions based on data. In contrast, the sketches in row 2 of Fig. 1 do not give enough information to discriminate between Fig. 1 *A* and *B*.

The purpose of this paper is to present a particular implementation of a set of algorithms that allows a systematic realization of all of these steps—inferring "normal forms" from data using just the data-mining counterpart of the above discussion: similarity between nearby observations/measurements. In row 3 of Fig. 1, we plot several simulated trajectories from a uniform grid of initial conditions in state space. Observing these data, one can easily infer the dynamical regime and deduce the underlying one-parameter family in a similar manner to the exercise above. Our methodology aspires to accomplish much more than what one can deduce based on such "easy" observations. For example, in row 4 of Fig. 1, we present the same type of data as in row 3 of Fig. 1 but depict far shorter trajectories. Now, the four images look much more similar—without prior knowledge or additional postprocessing, it would be challenging to correctly infer the dynamical regime and the one-parameter family from observations. Our methodology will be successful even with these short trajectory data.

In the language of systems theory, we consider multiple measurements at different parameter settings—what we call different trials from an unknown, nonlinear, parametrically dependent dynamical system. The problem setting comprises a large ensemble of short time series indexed by the label of the trial as well as by the label of the measurement channel; each time series is parametrized by time. However, the knowledge of how many and what parameters the system has and the actual settings at which the trials are performed is hidden. Furthermore, we do not know how many and what state variables the system has or what functions of the state variables we measure. We only know what each channel recorded, at each trial, as a function of time (for a short time). Using the similarity between individual pairs of this large ensemble of short time series as our only tool, we empirically build a normal form of the system: we identify a set of relevant parameters and a set of relevant state variables and help reveal the nature and relation of the associated phase portraits.

Recovering the underlying structure of nonlinear dynamical systems from data ("system identification") has attracted significant research efforts over many years, and several ingenious techniques have been proposed to address different aspects of this problem. These include methods to find nonlinear differential equations (1, 2) and to discover governing equations from time series or video sequences (3–5), equation-free modeling approaches (6), and methods for empirical dynamic modeling (7), just to name a few.

In this paper, we present a technique with roots in manifold learning (8–11), which involves diffusion geometry learning (12)

Significance

The extraction of models from data (in a sense, the "understanding" of the physical laws giving rise to the data) is a fundamental cognitive as well as scientific challenge. We show a geometric/analytic learning algorithm capable of creating minimal descriptions of parametrically dependent unknown nonlinear dynamical systems. This is accomplished by the data-driven discovery of useful intrinsic-state variables and parameters in terms of which one can empirically model the underlying dynamics. We discuss an *informed* observation geometry that enables us to formulate models without first principles as well as without closed form equations.

Author contributions: O.Y., R.T., R.R.C., and I.G.K. performed research and wrote the paper.

Reviewers included: A.J.C., University of California, Berkeley; and G.S., Duke University.

The authors declare no conflict of interest.

See Commentary on page 9998.

[1]To whom correspondence may be addressed. Email: ronen@ee.technion.ac.il, coifman@math.yale.edu, or yannis@princeton.edu.

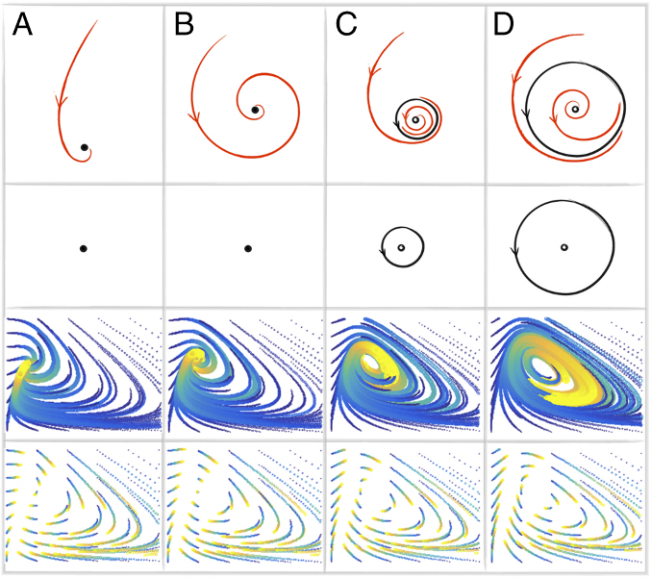

Fig. 1. Inferring a Hopf bifurcation from phase portrait representations (in the text). *A* and *B* consist of a single steady state, whereas *C* and *D* consist of a steady state and a stable limit cycle, depending on some critical parameter value.

and tensor geometry learning (13–15) as well as metric learning and approximation (16). Representation of dynamics in terms of diffusion geometries (and the "ergodic quotient") has been discussed in refs. 17–19. In our work, we present a simultaneous organization of observations originating from many different types of dynamical systems into a joint coherent structure, which parametrizes the various dynamical regimes and builds an empirical model of the whole observation space. In addition, we incorporate a variant of the earth mover's distance (EMD), which goes beyond classical transforms in establishing data-driven comparisons between trajectories.

Problem Formulation and a Toy Example

Our purpose is to devise a data-driven framework for the organization of time-dependent observations of dynamical systems depending on parameters. In our setting, these time-dependent measurements are the result of a number of experiments that we will call trials; during each trial, the (unknown) parameter values remain constant. In this black box setting, the dynamical system is unknown, nonlinear, and autonomous, and it is given by

$$\frac{d\boldsymbol{x}}{dt} = f(\boldsymbol{x};\boldsymbol{p}) \qquad [1]$$

$$\boldsymbol{y} = h(\boldsymbol{x}). \qquad [2]$$

We do not have access to its state $\boldsymbol{x}$ or to its parameter values $\boldsymbol{p}$; we also do not know the evolution law f or the measurement function h. We only have measurements (observations) $\boldsymbol{y}$ labeled by time t. The black box is endowed with "knobs" that, in an unknown way, change the values of the parameters $\boldsymbol{p}$; therefore, in every trial, for a new, but unknown, set of parameter values $\boldsymbol{p}$, we can observe $\boldsymbol{y}$ coming out of the box without knowing $\boldsymbol{x}$, f, or even h. We want to characterize the system dynamics by systematically organizing our observations (collected over several trials) of its outputs.

More specifically, we want to (*i*) organize the observations by finding a set of state variables and a set of system parameters that jointly preserve the essential features of the dynamics and then, (*ii*) find the corresponding intrinsic geometry of this combined variable–parameter space, thus building a sort of normal form for the problem. Small changes in this jointly intrinsic space will correspond to small changes in dynamic behavior (i.e., to robustness). Having discovered a useful "joint geometry," we can then inspect its individual constituents. Inspecting, for example, the geometry of the discovered parameter space will help identify regimes of different qualitative behavior. This might be different dynamic behavior, like hysteresis, or oscillations, separated by bifurcations; alternatively, we might observe transitions between different sizes of the minimal realizations: regimes where the number of minimal variables/parameters necessary in the realization changes.

We can also inspect the identified state variable geometry, which will help us organize the temporal measurements in coherent phase portraits. In addition, if there exist regimes where the system becomes singularly perturbed, we expect that we will be able to realize that the requisite minimal phase portrait dimension changes (reduces) and that the reduction in the number of state variables is linked with the reduction in the number of intrinsic parameters.

As an illustrative example, consider the following dynamical system, arising in the unfolding of the Bogdanov–Takens singularity (20):

$$\begin{aligned}\frac{dx_1}{dt} &= x_2\\ \frac{dx_2}{dt} &= \beta_1 + \beta_2 x_1 + x_1^2 - x_1 x_2.\end{aligned} \qquad [3]$$

This set of differential equations defines a dynamical system with two parameters $\boldsymbol{p} = (\beta_1, \beta_2)$, two state variables $\boldsymbol{x} = (x_1, x_2)$, and two observables $\boldsymbol{y} = (y_1, y_2)$; at first, we choose the observables to be the state variables themselves [i.e., $(y_1, y_2) = (x_1, x_2)$], with $h(\boldsymbol{x})$ being the identity function. It is known that the parameter space of this system (β_1, β_2) can be divided into four different regimes separated by one-parameter bifurcation curves (20). Fig. 2, *Lower Left* shows this "ground truth" bifurcation diagram for our simulated 2D grid of parameter values. Each point $\boldsymbol{p} = (\beta_1, \beta_2)$ on the grid is colored according to its respective dynamical regime.

Our goal in this case would be to discover an accurate bifurcation map of the system in a data-driven manner purely from observations. These observations consist of several samples, where each sample is a single trajectory $\boldsymbol{y}(t)$ of the system

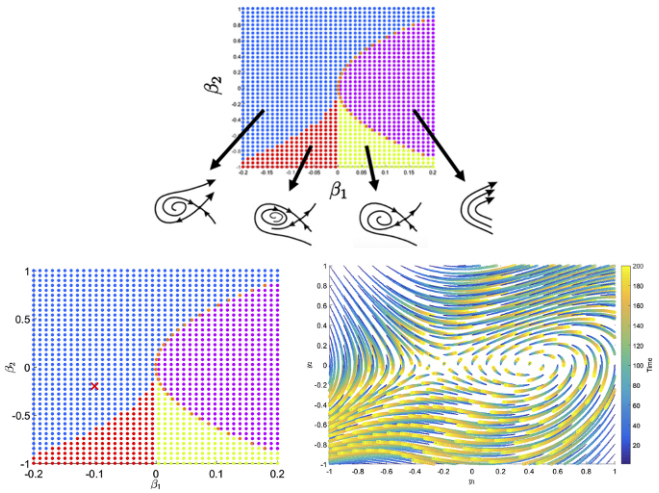

Fig. 2. (*Upper*) The Bogdanov–Takens bifurcation maps, with *Insets* illustrating the typical phase portraits in each dynamical regime. (*Lower Left*) The Bogdanov–Takens bifurcation map. (*Lower Right*) An example of the phase portrait of the simulated trajectories of the Bogdanov–Takens system corresponding to the parameter set $(\beta_1, \beta_2) = (-0.1, -0.2)$, marked by red X in *Lower Left*. The color and the width of the point correspond to time.

initialized with unknown (possibly different) parameter values and initial values. In addition, we would like to deduce from these large numbers of realizations of trajectories $\boldsymbol{y}(t)$ arbitrarily and differently initialized that the system depends on only two parameters and can be realized with only two state variables and to reconstruct the bifurcation diagram with its phase portraits.

Geometry Learning of Dynamics from Observations

Consider data arising from an autonomous dynamical system; we view the observations as entries in a 3D tensor. One axis of the tensor corresponds to variations in the problem parameters, one axis corresponds to variations in the problem variables, and the third axis corresponds to time evolution along trajectories.

Formally, let $\mathcal{P}$ denote an ensemble of N_p sets of the d_p system parameters. Let $\mathcal{V}$ be an ensemble of N_v sets of initial condition values of the d_v state variables. For each $\boldsymbol{p} \in \mathcal{P}$ and $\boldsymbol{v} \in \mathcal{V}$, we observe a trajectory $Y(\boldsymbol{v},\boldsymbol{p},t)$ of length N_t in $\mathbb{R}^{d_v}$ of the system variables, where $t = 1,\ldots,N_t$ denotes the time sample. In summary, $\boldsymbol{p}$ is a label of the particular trial (the particular differential equations) of the dynamical system, $\boldsymbol{v}$ is a label of the observations trajectory, and t is the time label.

Let $\mathbf{Y}$ denote the entire 3D tensor of observations of dimension $N_p \times N_v \times N_t$ consisting of all of the data at hand. With respect to the black box setting described in *Introduction*, we emphasize that the identity of the parameters and variables is hidden; we only have trajectories of observations corresponding to various trials with possibly different hidden parameter values and with different hidden initial input coordinates.

To make the problem definition concrete, we describe the setting of a specific example. Recall the Bogdanov–Takens dynamical system of two variables and two parameters introduced in Eq. **3**. We generate a set $\mathcal{P}$ of $N_p = 400$ different parameter values $\boldsymbol{p} = (\beta_1,\beta_2)$ from a regular fixed 2D grid, where $\beta_1 \in [-0.2, 0.2]$, $\beta_2 \in [-1, 1]$, and an additional 10 parameter values are located exactly on the bifurcation. Similarly, we generate a set $\mathcal{V}$ of $N_v = 441$ different initial conditions $\boldsymbol{v} = (y_1(0), y_2(0))$ from a fixed 2D grid in $[-1, 1]^2$. For each $\boldsymbol{p} \in \mathcal{P}$ and $\boldsymbol{v} \in \mathcal{V}$, we observe a trajectory of the system for $N_t = 200$ time steps, where the interval between two adjacent time samples is $\Delta t = 0.004$ [s], and collect all of the trajectories into a single 3D tensor $\mathbf{Y}$. In this example, $N_p = 410$, $N_v = 441$, and $N_t = 200$, and therefore, overall, we have $\mathbf{Y} \in \mathbb{R}^{410\times441\times200}$. For illustration purposes, Fig. 2, *Lower Right* depicts the phase portrait of all of the simulated 2D trajectories from all of the initial points for a single particular fixed value of $\boldsymbol{p}$, $\beta_1 = -0.1$, and $\beta_2 = -0.2$ (marked by a red X in Fig. 2, *Lower Left*). We note that the trajectories (as illustrated in Fig. 2) are long enough to partially overlap in phase space. Such an overlap induces the coupling between the time and variables axes, which is captured and exploited by our analysis. We wish to find a reliable representation of the hidden parameters, of the hidden variables, and of the time axis.

Define $\boldsymbol{y}_p = \{Y(\boldsymbol{v},\boldsymbol{p},t)|\forall \boldsymbol{v},\forall t\}$ for each of the N_p vectors of hidden parameter values $\boldsymbol{p}$ in $\mathcal{P}$, namely, a data sample consisting of all of the trajectories from a single trial. For simplicity of notation, we will use subscripts to denote both the appropriate axis and a specific set of entries values on the axis. We refer to $\{\boldsymbol{y}_p\}, \boldsymbol{p} \in \mathcal{P}$ as the data samples from the parameters axis viewpoint. In the Bogdanov–Takens example, Fig. 2, *Lower Right* depicts $\boldsymbol{y}_p$ for $\boldsymbol{p} = (\beta_1,\beta_2) = (-0.1, -0.2)$.

Similarly, let $\boldsymbol{y}_v$ and $\boldsymbol{y}_t$ be the samples from the viewpoints of the variables axis and the time axis, respectively, which are defined by

$$\boldsymbol{y}_v = \{Y(\boldsymbol{v},\boldsymbol{p},t)|\forall \boldsymbol{p},\forall t\}, \quad \boldsymbol{v} \in \mathcal{V}$$
$$\boldsymbol{y}_t = \{Y(\boldsymbol{v},\boldsymbol{p},t)|\forall \boldsymbol{v},\forall \boldsymbol{p}\}, \quad t = 1,\ldots,N_t.$$

One way to accomplish our goal is to process the data three successive times, each time from a different viewpoint.

Here, we use a data-driven parametrization approach based on a kernel. From the trials (effectively, parameters) axis point of view, a typical kernel is defined by

$$k(\boldsymbol{y}_{p_1},\boldsymbol{y}_{p_2}) = e^{-\frac{\|y_{p_1}-y_{p_2}\|^2}{\epsilon}}, \forall \boldsymbol{p}_1,\boldsymbol{p}_2 \in \mathcal{P} \qquad [4]$$

based on distances between any pair of samples, where the Gaussian function induces a sense of locality relative to the kernel scale ϵ. To aggregate the pairwise affinities comprising the kernel into a global parametrization, traditionally, the eigenvalue decomposition (EVD) is applied to the kernel, and the eigenvalues and eigenvectors are used to construct the desired parametrization. The specific parametrization method that is used here is diffusion maps (12), which is described in detail in *Appendix 1: Diffusion Maps*.

From three separate diffusion maps applications to the sets $\{\boldsymbol{y}_p\}$, $\{\boldsymbol{y}_v\}$, and $\{\boldsymbol{y}_t\}$, we can obtain three mappings as in Eq. **9**, denoting the associated eigenvectors by $\{\boldsymbol{\psi}_\ell^{\mathcal{P}}\}$, $\{\boldsymbol{\psi}_\ell^{\mathcal{V}}\}$, and $\{\boldsymbol{\psi}_\ell^{\mathcal{T}}\}$, respectively.

However, such mappings do not take into account the strong correlations and codependencies between the parameter values and the dynamics of the variables, which arise in typical dynamical systems. For example, in the Bogdanov–Takens system, the dynamical regime changes significantly depending on the values of the parameters.

To incorporate such codependencies, we propose to define and build from observations an informed metric between samples in the different axes. In the introduction of the affinity matrix in Eq. **4**, we deliberately did not specify the norm used to compare between two samples. Common practice is to use the Euclidean norm. However, as pointed out by Lafon (21), anisotropic diffusion maps can be computed by using different norms. This issue has been extensively studied recently, and several norms and metrics have been developed for this purpose (22–24).

Here, following refs. 13–15 and 25, we propose a particular construction of an informed metric that relies on the geometry of the coordinates of the samples; the metric and the (induced) geometry evolve together, as will be described below. For simplicity, the exposition begins by focusing on the analysis of the data from the perspective of the parameters axis; the generalization to the other two axes is analogous.

The essence of our analysis is the definition of a meaningful notion of distance between the samples. Specifically, we build an informed distance metric $\|\boldsymbol{y}_{p_1}-\boldsymbol{y}_{p_2}\|_{\mathcal{P}}$, where the subscript of the norm $\mathcal{P}$ indicates that it is an informed norm between the samples—informed from the parameters viewpoint. The construction of the metric is implemented in an iterative manner. The idea is that, in each iteration, the codependencies between the axes are gradually revealed from observations and in turn, used to build a refined informed metric. The full details of the construction procedure are presented in *Appendix 2: Informed Metric Construction*. Here, we only describe the first iteration when applied to data arising from the Bogdanov–Takens system.

In the first iteration, the construction of the informed metric $\|\cdot\|_{\mathcal{P}}$ defined on the parameters axis uses as an initial input two noninformed metrics, defined on the variables axis and on the time axis, with roles that will be made clear in the sequel. Possible choices for such metrics are the Euclidean metric or a metric derived from the cosine similarity.

The construction itself is implemented by decomposing the metric into the following general form:

$$\|\boldsymbol{y}_{p_1}-\boldsymbol{y}_{p_2}\|_{\mathcal{P}} = \|\boldsymbol{y}_{p_1}-\boldsymbol{y}_{p_2}\|_1 + \gamma\|\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_{p_1})-\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_{p_2})\|_1, \qquad [5]$$

where $\|\cdot\|_1$ is the ℓ_1 norm, $\gamma > 0$ is a positive weighting factor, and $\mathcal{F}_{\mathcal{P}}\colon \mathbb{R}^{N_v\times N_t} \to \mathbb{R}^{D_p}$ is some feature function (to be discussed). We note that the particular choice of the ℓ_1 norm is

explained in detail in ref. 16 and will be reviewed in *Appendix 2: Informed Metric Construction*; however, other norms can be used depending on the application at hand. The function $\mathcal{F}_{\mathcal{P}}$ is, therefore, viewed as a generalized transform of the samples, and the problem of finding a meaningful metric is transformed to the problem of finding an appropriate transformation $\mathcal{F}_{\mathcal{P}}$. In this work, we present a transform that appends coordinates to the samples, such that the ℓ_1 norm is equivalent to a generalized EMD. The EMD is a measure of distance between probability distributions, which is also known as the Wasserstein distance (26, 27). Informally, given two piles of earth (dirt), it can be viewed as the minimal cost of turning (moving) one pile into the other. This metric is commonly used for comparing images and probability density functions (26). Recently, efficient computation techniques (28) and generalizations (16, 29) were presented, which do not require a direct computation of the cost but rather, use generalized transforms as in Eq. **5**. Because the EMD (in contrast, for example, to the Euclidean distance) takes into account the "ground distance" (26), it is stable under small deformations of the data. This property is important in the context of dynamical systems, as it allows for the comparison of similar trajectories even when partially overlapping or when the trajectories are long and the norm between them diverges.

Traditional (2D) transforms are typically implemented using a set of basis functions $g_{\ell,\ell'}$, and the transform is generally given by a collection of the linear projections of the data on that set of basis functions:

$$\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p) = \left\{\langle g_{\ell,\ell'}, \boldsymbol{y}_p\rangle | \forall \ell\right\}. \quad \textbf{[6]}$$

For the parameters axis, the basis functions $g_{\ell,\ell'}$ are defined on $\mathcal{V} \times \{1,\ldots,N_t\}$, and the inner product is given by

$$\langle g_{\ell,\ell'}, \boldsymbol{y}_p\rangle = \sum_{v\in\mathcal{V}} \sum_{t=1,\ldots,N_t} g_{\ell,\ell'}(\boldsymbol{v},t)\, Y(\boldsymbol{v},\boldsymbol{p},t).$$

Such classical transforms include the Fourier Transform, the Wavelet transform, etc. However, these transforms are linear and local, and their basis functions $g_{\ell,\ell'}$ are fixed and are not data-adaptive.

To circumvent these limitations, following refs. 14 and 15, we propose a transform based on data-driven partition trees. By using the initial noninformed metrics on the variables and on the parameters, multilevel partitions of both the variables axis and the time axis are computed. In turn, these trees are used to define an overcomplete set of basis functions; a basis function is defined for each folder I_ℓ in the variables tree and for each folder $J_{\ell'}$ in the time axis tree as the indicator function for the samples in these folders, that is,

$$g_{\ell,\ell'}(\boldsymbol{v},t) = \begin{cases} 1 & \boldsymbol{v}\in I_\ell, t\in J_{\ell'} \\ 0 & \text{otherwise.}\end{cases}$$

After we have the basis functions, the transform can be formulated, and based on that, the EMD can be defined. Details are in *Appendix 2: Informed Metric Construction*.

Note that the partition trees provide a specific set of basis functions; however, any basis function set could be used for this purpose. A particular alternative implementation is via the eigenvectors of diffusion maps (12). Broadly, the informed metric is implemented by a generalized transform that appends coordinates to the original samples, such that a noninformed norm/distance between the resulting, augmented/transformed samples retains particularly desired properties. Because the outer products of the eigenvectors obtained by diffusion maps $\{\boldsymbol{\psi}_\ell^{\mathcal{V}} \otimes \boldsymbol{\psi}_{\ell'}^{\mathcal{T}}\}$ are defined on $\mathcal{V}\times\{1,\ldots,N_t\}$, they can be used as a set of basis functions of $\{\boldsymbol{y}_p\}$, and additional coordinates can be appended to the samples by projections on these functions. This alternative implementation is being currently explored.

Several remarks are due at this point. First, the recursive procedure described above repeats in iterative manner, where in each iteration, three informed metrics are constructed one by one based on the metrics from the preceding iteration. As the iterations progress, the metrics are gradually refined, and the dependency on the initialization is reduced. The outline of the recursive algorithm is presented in *Algorithm 1*. Similar iterative approaches for decomposing 3D tensors are based on iterative singular value decompositions obtained by isolating an axis (30). Second, the characterization of the stopping criterion and the convergence of this iterative procedure should be possible through the combination of asymptotic analysis of the informed kernels (12) with the alternating ℓ_1 minimization in refs. 31 and 32. Currently, we typically apply a few iterations (in this paper, up to two); this has empirically led to good performance. Third, the exposition here focuses on transforms based on linear projections on basis functions. We note that nonlinear embedding constructed from the basis functions themselves [e.g., diffusion maps (12)] can also be used as the additional coordinates appended to the samples. This point, as well as additional technical details, will be further discussed in *Appendix 2: Informed Metric Construction*.

Examples

The Matlab code used in the following examples is available online at https://github.com/oryair/InformedGeometry-CoupledPendulum.

Bogdanov–Takens Bifurcation Mapping. Our method is applied to the 3D tensor of trajectories $\mathbf{Y}$ collected from the Bogdanov–Takens system. As described above, $\mathbf{Y}$ consists of (short) trajectories of observations arising from the system initialized with various initial conditions and with various parameters. We emphasize that the knowledge of the different regimes and the bifurcation map was not taken into account in the analysis; only the time-dependent data $\mathbf{Y}$ were considered.

Fig. 3*A* depicts the scatter plot of the two dominant eigenvectors representing the parameters axis. It consists of N_p points (the length of the eigenvectors), where each point corresponds to a single sample $\boldsymbol{y}_p \in \mathbb{R}^{N_v\times N_t}$, which is associated with parameters values $\boldsymbol{p} = (\beta_1, \beta_2)$ on the 2D grid depicted in Fig. 2. Moreover, each point in Fig. 3*A* is colored by the same color-coding used in Fig. 2.

We observe that our method discovers an empirical bifurcation mapping of the system. Indeed, the obtained representation of the parameters through the eigenvectors establishes a coordinate system with a geometry built solely from observations, which reflects the organization of the parameters space according to the true underlying bifurcation map—the "visual homeomorphism" (stopping short of claiming visual isometry) is clear.

To illustrate the generality of our method, we now apply a nonlinear (yet invertible) observation function

$$\boldsymbol{z}(t) = h(\boldsymbol{x}(t))$$

with $h_k(\boldsymbol{x}(t)) = \sqrt{\boldsymbol{a}_k^T \boldsymbol{x}(t) + \alpha_k}$, $k = 1, 2$, where $\boldsymbol{a}_k$ is a random observation vector, and α_k is a constant set to guarantee positivity. Fig. 3*B* depicts the scatter plot of the two dominant eigenvectors representing the parameters axis obtained from the new set of nonlinear observations. A visibly equivalent organization is clearly achieved.

Fig. 3*C* depicts the scatter plot of the two dominant eigenvectors representing the state variable axis. The plot consists of N_v points (the length of the eigenvectors), where each point corresponds to a single sample $\boldsymbol{y}_v \in \mathbb{R}^{N_p\times N_t}$, which is associated with a particular set of initial condition values $\boldsymbol{v} = (y_1(0), y_2(0))$. The embedded points are colored in Fig. 3*C* by the initial conditions of the variable y_1 and in Fig. 3*D* by the initial conditions

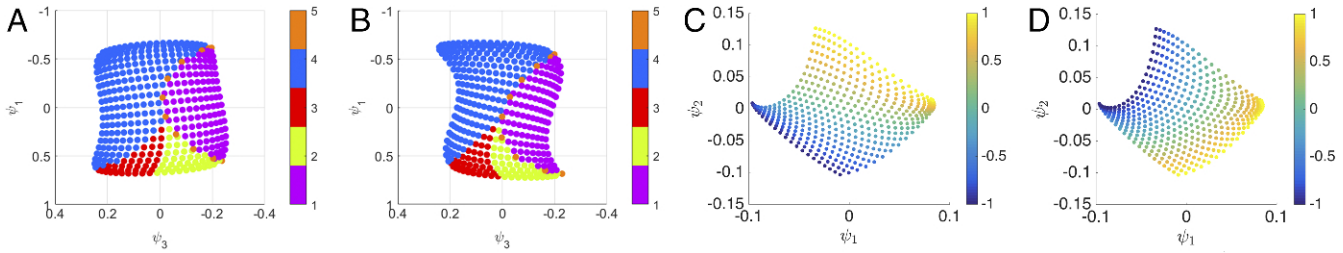

Fig. 3. (*A* and *B*) Data-driven embedding of the parameters axis of the observations collected from the Bogdanov–Takens system (colored according to the true bifurcation map). Embeddings built from (*A*) state variable observations and (*B*) observations through a nonlinear invertible function. (*C* and *D*) Data-driven embedding of the state variables axis colored by (*C*) the initial conditions of x_1 and (*D*) the initial conditions of x_2.

of the variable y_2. The color-coding implies that the recovered 2D space corresponds to (is one-to-one with) the 2D space of the true variables of the system. In other words, the high-dimensional samples $\boldsymbol{y}_v$ are embedded in a 2D space, which recovers a 2D structure accurately representing the true directions of the hidden, minimal, two-state variables of the system.

Two Coupled Pendula. We simulate a system of two simple coupled pendula with equal lengths L and equal masses m connected by a spring with constant k as shown in Fig. 4, *Left*. Let $u^{(1)}(t)$ and $u^{(2)}(t)$ denote the horizontal positions of the pendula, and let $\theta^{(1)}(t)$ and $\theta^{(2)}(t)$ be the angles between each of them and the vertical axis. Note that a closed form expression for the positions cannot be derived for the general case. However, in the case of small perturbations around the equilibrium point, we consider a linearized regime and assume that $\sin\theta^{(i)} \approx \theta^{(i)}$ for $i = 1, 2$. We present the linear case, because it can be trivially reproduced and thus, validated; however, our data-driven approach can equally well be performed for the more general nonlinear case. The ordinary differential equation (ODE) system describing the evolution of the horizontal positions within this linear regime is given by

$$\begin{cases} m\ddot{u}^{(1)} = -\dfrac{mg}{L}u^{(1)} - k\left(u^{(1)} - u^{(2)}\right) \\ m\ddot{u}^{(2)} = -\dfrac{mg}{L}u^{(2)} + k\left(u^{(1)} - u^{(2)}\right) \end{cases}, \qquad [7]$$

where $\ddot{u}$ is the second derivative of u, and g represents gravity. To highlight the broad scope of our approach from a data analysis perspective, we assume that we do not have direct access to the horizontal displacement. Instead, we generate movies of the motion of the coupled pendula (observed in the form of pixels; see below) in the linear regime. On the one hand, we have a definitive, physical law-based ground truth described by the solution of the ODE of the system (in terms of the two normal modes of the system) (details are in *Appendix 3: More on the Two Coupled Pendula*). On the other hand, we only have access to high-dimensional nonlinear observations of the system: the pixels of the frames of the movie.

In the first experiment, we generate three movies corresponding to three different spring constants k_p, where $p = 1, 2, 3$. As a consequence, the motion in each movie consists of two oscillations:

$$\omega_1 = \sqrt{\frac{g}{L}}, \quad \omega_p = \sqrt{\frac{g}{L} + \frac{2k_p}{m}},$$

where $L = 1$, $m = 1$, $g = 9.8$, and $k_p \in \{750, 900, 1050\}$. Each movie comprises 400 frames and is 4 s long. Each frame of the movie consists of 20×40 black, white, red, green, and blue pixels; an example of one movie can be found at https://youtu.be/rjUxeT-ShQc. Our data are, therefore, $\boldsymbol{Y} \in \mathbb{R}^{N_p \times N_v \times N_t}$, where $N_p = 3$, $N_v = 800$, and $N_t = 400$. We apply *Algorithm 1* to the input data $\mathbf{Y}$ (we could apply it to the digitization of an actual experiment). Fig. 4, *Center* displays the Fourier transform of the most dominant eigenvector from the time embedding. It is clearly seen that all of the frequencies ω_1 and ω_p are captured, where the fixed frequency ω_1 is dominant and the varying-across-movies frequencies ω_p are less pronounced as expected. This experiment shows the features of our methodology, and the empirical results are presented in light of the known ground truth.

In the second experiment, we simulate a system with time-varying spring constant $k(t)$. In this case, we do not have an analytic solution for any $k(t)$ (*Appendix 3: More on the Two Coupled Pendula*), and therefore, there is no definitive ground truth; however, we show that our method enables us to recover two normal modes even in this (explicitly unsolvable) scenario. Now, we generate a movie, where the spring constant $k(t)$ changes over time

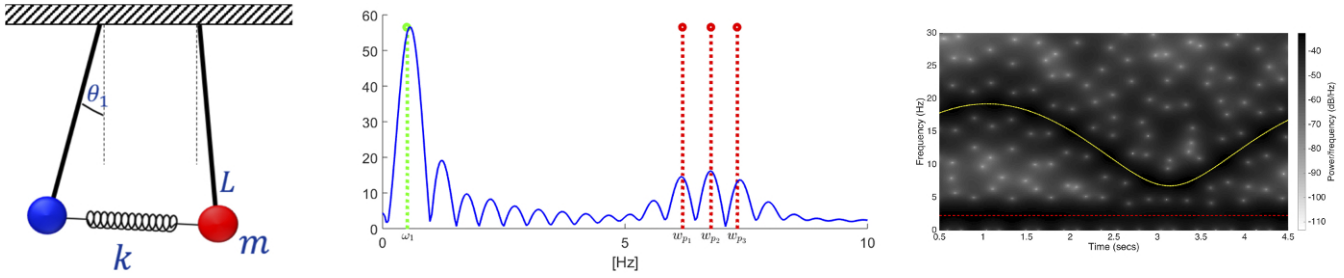

Fig. 4. (*Left*) Schematic of the two coupled pendula. (*Center*) The Fourier transform of the principal eigenvector representing the time axis (blue) obtained in the first experiment. The dashed lines are the true (hidden) frequencies. The dashed green line corresponds to the fixed oscillation frequency ω_1, and the dashed red lines correspond to the varying-across-movies oscillation frequency ω_p. (*Right*) The Fourier spectrogram of the principal eigenvector representing the time axis obtained in the second experiment. The two oscillation frequencies ω_1 and $\omega_2(t)$ are overlaid on the spectrogram. The dashed red line corresponds to the fixed oscillation frequency ω_1, and the dotted yellow line corresponds to the time-varying oscillation frequency $\omega_2(t)$.

according to $k(t) = 1{,}000 + 800\sin(1.5t)$. This movie is available at https://www.youtube.com/watch?v=I8C6Yt3b2tk.

In the context of our data processing, the variables axis consists of the 800 pixels, and the time axis represents the 400 frames. To simplify the exposition, we focus here on a *single* parameter setting, so that the parameters axis is degenerate (i.e., $N_t = 400$, $N_v = 800$, $N_p = 1$, and $\mathbf{Y} \in \mathbb{R}^{800\times 400}$).

We apply our method to the data from the simulated movie. Fig. 4, *Right* depicts the Fourier spectrogram of the principal eigenvector representing the time axis obtained by diffusion maps with the informed metric after one iteration. We observe that the empirical representation of the time axis identifies two oscillation frequencies $\omega_1 = \sqrt{g/L}$ and $\omega_2(t) = \sqrt{g/L + 2k(t)/m}$, which are emphasized by overlaid curves.

To highlight the scope and potential of our approach, we now apply a fixed, invertible, random projection to each frame of the movie. In other words, each frame of the movie was multiplied by a fixed matrix, with columns that were independently sampled from a multivariate normal distribution and normalized to have a unit norm. The resulting movie with the projected frames can be found at https://www.youtube.com/watch?v=xz0hzQTyPGo. Fig. 5 depicts an example of three snapshots of the coupled pendulum system paired with their random projection counterparts. The obtained parametrization, representing the time axis obtained from the new "scrambled" movie, is presented in Fig. 6, where we observe that the same two frequencies ω_1 and $\omega_2(t)$ are captured by our method, despite the additional unknown "observation" function. An additional experiment appears in *Appendix 3: More on the Two Coupled Pendula*.

Summary and Discussion

Obtaining predictive dynamical equations from data is at the heart of science and engineering modeling and is the linchpin of our technology. Today, we witness the development of mathematical techniques that operate directly on observations (i.e., data-driven) and appear to circumvent the need to "manually" select variables and parameters and to derive accurate equations. The core of this methodology has shifted to the development of the mathematics and algorithms that systematically transition from data to the analysis of the model and thus, to making predictions, without ever deriving the model in closed form. Our work here presents an illustration of this path in an attempt to extend classical transformative results of linear system identification (33) and realization theory (34, 35). Note that we assumed that we had plenty of data, whereas the questions about the fewest measurements required for modeling and how to efficiently plan and conduct their sampling have been left open for future work.

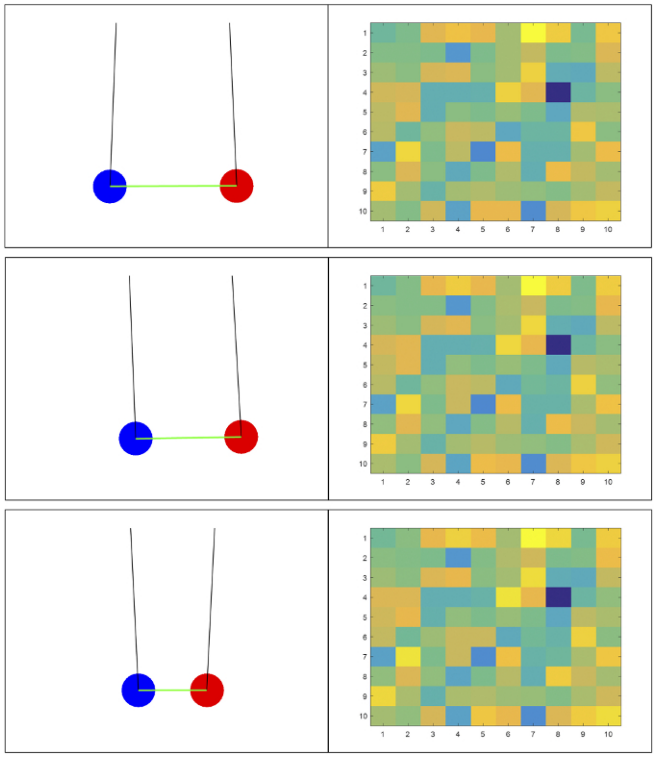

Fig. 5. An example of three snapshots of the coupled pendulum system paired with their random projection counterparts.

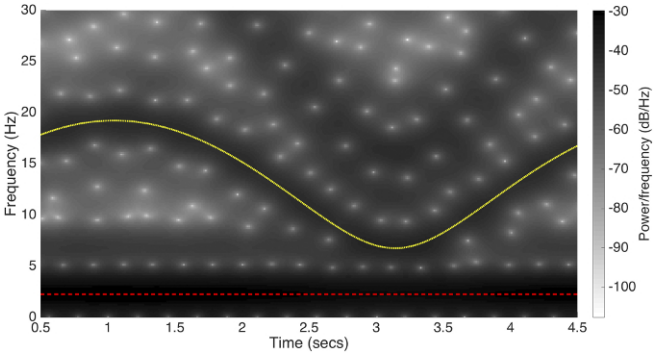

Fig. 6. The Fourier spectrogram of the principal eigenvector representing the time axis. These results are based on the random projections of the movie frames with the same time-varying spring constant. The two frequencies ω_1 and $\omega_2(t)$ are overlaid on the spectrogram. The dashed red line corresponds to the fixed oscillation frequency ω_1, and the dotted yellow line corresponds to the time-varying oscillation frequency $\omega_2(t)$.

We have shown that (invertible) functions of our measurements give rise to homeomorphic (and possibly isometric) embeddings, thereby conveying the same prototypical behavior [in the spirit of Whitney and the even stronger Nash embedding theorems and more specifically, Takens embedding for dynamical systems (36–39)]. This points toward the feasibility of "gauge-invariant" data mining (22, 24, 40–42) through algorithms that do not depend on the measurement modality.

While we only use indices for the trials and the measurement channels, we did keep the sequential parametrization of the measurements by time (here, at equal time intervals); this is not necessary, and one could use only labels also in the time axis (denoting what measurements were obtained at the same moment in time but without knowing the actual time stamp). This would lead to the correct ordering of the measurement snapshots without providing actual time stamps for them (43). More generally, semisupervised learning and "manifold completion" tools can be used to fruitfully fill in missing data, interpolate, and (modestly) extrapolate the input–output functions learned here. Additionally, while our approach has been built to directly apply to real world data, it would be particularly exciting to apply it as a second "debriefing" layer to create minimal, balanced realizations of the very high-dimensional representations of such data learned by a first reconnaissance by deep nets (43).

In learning features of an input–output description, we provided the algorithm with data that we knew were "relevant." How would the algorithm perform if we presented it with some extraneous data along with the relevant ones? This subject, which is crucial for learning, is addressed in ref. 44, where data that can be deduced to be measurements of the same process can be systematically identified. It is also clear that the conditions for convergence of our "triple successive iteration" scheme (what one might call a "consistent input–output

balancing/renormalization") should be established mathematically. To this end, the seminal work of Breiman and Friedman (45) will clearly be useful in this effort (31), while also laying the ground for statistical and possibly Bayesian interpretations.

While our construction takes into account the coupling between state space, parameters, and time, our final representations for each "entity" are separate. Perhaps the most natural extension of this work is in the direction of detecting changes in the state or parameter embedding dimensions. Whether distinct (regular perturbations) or joint (singular perturbations), these dimension changes are crucial for successful model reduction. For this purpose, in future work, we will examine the expansion of the high-dimensional data in the joint parameters–state–time space obtained by a product of the inferred representation components. This will establish causality and provide an intrinsic form of the governing equations of the dynamical system.

Appendix 1: Diffusion Maps

One class of data-driven methods for analyzing complex datasets is manifold learning. The main assumption in manifold learning is that, often, data are constrained to lie on or around a low-dimensional manifold. The idea is then to build a low-dimensional embedding of the data in a data-driven way, such that it parametrizes the underlying manifold structure. A particular manifold learning method, diffusion maps, is used as a building block in our approach and briefly described next.

Given a set $\{\boldsymbol{y}_t\}$ of N_t samples of observations of the system, let $\mathbf{W}$ be an $N_t \times N_t$ pairwise affinity matrix, with the (t,s)th entry that is defined by

$$W_{t,s} = e^{-\frac{\|\boldsymbol{y}_t - \boldsymbol{y}_s\|^2}{\epsilon}}, \quad [8]$$

where $\epsilon > 0$ is a positive scale parameter. Common practice is to interpret the set of samples and the affinity matrix as a graph, where the samples are the graph nodes, and the affinity matrix determines the weights of the edges (i.e., node $\boldsymbol{y}_t$ is connected to node $\boldsymbol{y}_s$ by an edge with weight $W_{t,s}$). Setting the appropriate value of the kernel scale ϵ has been the subject of many studies (46, 47). One standard approach is to set it as the median of all of the distances $\|\boldsymbol{y}_t - \boldsymbol{y}_s\|^2$ in $W_{t,s}$; in the graph interpretation, this results in well-connected graphs and therefore, enables the user to take into account a large number of relationships between the available samples.

The next step aggregates the pairwise affinities/graph connections into a global parametrization. Before that, the affinity matrix is traditionally normalized; several normalizations are considered in the literature, and here, we present a particularly common one. Let $\mathbf{D}$ be a diagonal matrix with main diagonal that comprises the sum of rows of $\mathbf{W}$, and let $\mathbf{A} = \mathbf{D}^{-1}\mathbf{W}$ be a row-stochastic matrix. In the graph interpretation, $\mathbf{A}$ can be seen as a transition probability matrix defining a Markov chain on the graph, where $A_{t,s}$ is the probability to "jump" from node $\boldsymbol{y}_t$ to node $\boldsymbol{y}_s$ in one Markov chain step.

The global parametrization is obtained by applying the EVD to $\mathbf{A}$. Let λ_ℓ denote the eigenvalues, ordered in decreasing order, and let $\boldsymbol{\psi}_\ell$ denote the corresponding (right) eigenvectors. Note that $\mathbf{A}$ is row stochastic, and hence, its largest eigenvalue is $\lambda_0 = 1$, corresponding to an all-ones trivial eigenvector $\boldsymbol{\psi}_0$; since both do not carry information on the data, they are typically ignored. The diffusion maps embedding of the samples is defined as the following nonlinear map for some $\tau > 0$:

$$\boldsymbol{y}_t \mapsto (\lambda_1^\tau \psi_1(t), \lambda_2^\tau \psi_2(t), \ldots, \lambda_m^\tau \psi_m(t)), \quad [9]$$

where each sample $\boldsymbol{y}_t$ is embedded in $\mathbb{R}^m$ by the values of the m eigenvectors associated with the m largest eigenvalues.

The diffusion maps embedding bears two important properties. First, it can be shown that the Euclidean distance between the embedded samples approximates the diffusion distance, a distance defined by the induced transition probabilities and that is closely related to the geodesic distance on the assumed underlying manifold (12). Second, the eigenvectors $\boldsymbol{\psi}_\ell$ form an orthonormal basis for any real function defined on the sample set $\{\boldsymbol{y}_t\}$. More details are in ref. 12.

Appendix 2: Informed Metric Construction

Partition Trees. We build a partition tree for each axis based on a given metric. Each partition tree is composed of $L+1$ levels, where a partition $\mathcal{I}_l$ of the samples is defined for each level $0 \le l \le L$. The partition $\mathcal{I}_l$ at level l consists of $n(l)$ mutually disjoint nonempty subsets of samples, termed folders and denoted by $I_{l,i}$, $i \in \{1, \ldots, n(l)\}$, that is,

$$\mathcal{I}_l = \{I_{l,1}, I_{l,2}, \ldots, I_{l,n(l)}\}. \quad [10]$$

The partition tree has the following properties:

- The finest partition ($l = 0$) is composed of singleton folders, termed the "leaves," where $I_{0,i} = \{\boldsymbol{v}_i\}$, and the number of leaves is the total number of samples.
- The coarsest partition ($l = L$) is composed of a single folder $I_{L,1}$, which is termed the "root" of the tree and consists of all of the samples.
- Each folder at level $l-1$ is a subset of a folder from level l (i.e., the partitions are nested such that, if $I \in \mathcal{I}_l$, then $I \subseteq J$ for some $J \in \mathcal{I}_{l+1}$).

The partition tree is the set of all folders at all levels:

$$\mathcal{T} = \{I_{l,i} \,|\, 0 \le l \le L,\ 1 \le i \le n(l)\}. \quad [11]$$

Ref. 15 has more details.

There are multiple ways to build such partition trees. The different construction methods can be divided into two classes: bottom-up construction and top-down construction. Broadly, a bottom-up construction begins with the definition of the lower levels, initially by grouping the leaves/samples (e.g., using k-means). Then, these groups are further grouped in an iterative procedure to create the next levels, ending at the root, in which all of the samples are placed under a single folder. A top-down construction is typically implemented by an iterative clustering method, initially applied to the entire set of samples and then refined over the course of the iterations, starting with the root of the tree and ending at the leaves.

Iterative Metric Construction. The construction of the partition tree described above relies on a metric between the samples. We propose a procedure, in which the construction of the tree relies on an iteratively evolving "informed metric" induced by partition trees on the coordinates of the samples. Namely, the construction of $\mathcal{T}_v$ relies on a metric between the samples $\boldsymbol{y}_v$, and the construction of $\mathcal{T}_t$ relies on a metric between the samples $\boldsymbol{y}_t$. Given $\mathcal{T}_v$ and $\mathcal{T}_t$, the informed metric between the samples $\boldsymbol{y}_p$ is constructed and then, used to build a partition tree $\mathcal{T}_p$ of the samples $\boldsymbol{y}_p$. In the second substep within the iteration, $\mathcal{T}_p$ can be used to construct refined metrics between $\boldsymbol{y}_v$ and between $\boldsymbol{y}_t$. In what follows, we describe one full iteration in detail.

Let $\mathcal{T}_v$ and $\mathcal{T}_t$ denote finite partition trees of the samples $\{\boldsymbol{y}_v\}$ and $\{\boldsymbol{y}_t\}$, respectively. The partition trees $\mathcal{T}_v$ and $\mathcal{T}_t$ induce a multiscale decomposition on the data, particularly of the coordinates of the samples $\{\boldsymbol{y}_p\}$. Here, we show how this decomposition is used to construct an informed metric between the samples $\{\boldsymbol{y}_p\}$ and in turn, a partition tree $\mathcal{T}_p$ on $\{\boldsymbol{y}_p\}$; we formulate it by the construction of a data-adaptive filter bank. Define the filter $g_{I\times J}$ for each $I \in \mathcal{T}_v$ and $J \in \mathcal{T}_t$ by

$$g_{I\times J} = \frac{w(I,J)}{|I||J|} \mathbb{1}_I \otimes \mathbb{1}_J, \quad [12]$$

where $w(I,J)$ are positive weights, $|\cdot|$ denotes the cardinality of a set, $\otimes$ denotes the outer product, and $\mathbb{1}_I$ and $\mathbb{1}_J$ denote the

indicator vectors of the samples in folder $I \in \mathcal{T}_v$ and folder $J \in \mathcal{T}_t$. For each filter, we calculate the inner product between the filter $g_{I\times J}$ induced by folders I and J and a sample $\boldsymbol{y}_p$, yielding a scalar coefficient $f_{I\times J}$:

$$f_{I\times J}(\boldsymbol{y}_p) = \langle g_{I\times J}, \boldsymbol{y}_p \rangle = \frac{w(I,J)}{|I||J|} \sum_{\boldsymbol{v}\in I, t\in J} Y(\boldsymbol{p}, \boldsymbol{v}, t). \quad [13]$$

By definition, $f_{I\times J}(\boldsymbol{y}_p)$ can be viewed as a nonlocal, informed low-pass filter, since it computes a nonlocal average of the coordinates of $\boldsymbol{y}_p$ prescribed by the folders I and J. The transform $\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p)$ is then defined by

$$\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p) = \left\{ f_{I\times J}(\boldsymbol{y}_p) | \forall I \in \mathcal{T}_v, \forall J \in \mathcal{T}_t \right\}.$$

In other words, $\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p)$ is the collection of all coefficients $f_{I\times J}(\boldsymbol{y}_p)$ resulting from the applications of all possible filters $g_{I\times J}$ to $\boldsymbol{y}_p$.

The main property of $\mathcal{F}_{\mathcal{P}}$ stems from the utilization of the codependencies of the variables and time samples in a data-driven manner; by combining mutually related, yet nonlocal, sets of variables and time samples, an informed representation of the samples from the parameters axis is obtained. In addition, $\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p)$ can be viewed as appending coordinates to $\boldsymbol{y}_p$ by computing averages according to the partition trees. A particular property of the ℓ_1 norm of these "vectors" $\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p)$ is described next.

We set the weights as

$$w(I,J) = \left(\frac{|I|}{N_v}\right)^{\beta_v+1} \left(\frac{|J|}{N_t}\right)^{\beta_t+1}, \quad [14]$$

where β_v weighs the bifolder $I \times J$ based on the relative size of folder $I \in \mathcal{T}_v$, and β_t weighs the bifolder based on the relative size of $J \in \mathcal{T}_t$. These values should be set according to the smoothness of the data along the respective axis; the smoother the data are along each axis, the larger the corresponding coefficient should be, thereby promoting larger folders (coarser representation). With these weights, the ℓ_1 distance between the transformations, that is,

$$\|\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p) - \mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_q)\|_1 \quad [15]$$

equals the EMD between the samples $\boldsymbol{y}_p$ (15). By definition, we have

$$\|\mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_p) - \mathcal{F}_{\mathcal{P}}(\boldsymbol{y}_q)\|_1 = \sum_{I\in\mathcal{T}_v, J\in\mathcal{T}_t} |f_{I\times J}(\boldsymbol{y}_p) - f_{I\times J}(\boldsymbol{y}_q)|. \quad [16]$$

This distance thus enables us, in an efficient data-driven manner, to compute the EMD based only on nonlocal averaging filters induced by the partition trees. After the metric is constructed, it can be used to build a partition tree $\mathcal{T}_p$ on the samples $\boldsymbol{y}_p$.

The construction of the informed metric between the samples $\boldsymbol{y}_p$ described above is repeated in an analogous manner to build informed metrics between the samples $\boldsymbol{y}_v$ and between the samples $\boldsymbol{y}_t$. The entire iterative construction procedure is summarized in *Algorithm 2*.

Finally, we remark on two possible variants of the algorithm. In this presentation, the construction of the tree is carried out by relying directly on the informed EMD. Alternatively, we can first apply diffusion maps to the samples based on the EMD and then use the diffusion distances to build the trees. This will introduce an additional nonlinear stage, in which the local information conveyed by the EMD is aggregated into a more global metric between the samples (realized by appending additional "informed" coordinates to the samples). The algorithm outline for this variant is given in *Algorithm 1*. A second variant relates to the definition of the filters $g_{I\times J}$, which are based on indicator functions $\mathbb{1}_I$ (tensor Haar basis) on the partition tree folders I. Instead, we can use the eigenvectors stemming from diffusion maps. Specifically, let $\boldsymbol{\psi}_\ell^{\mathcal{V}}$ and $\boldsymbol{\psi}_{\ell'}^{\mathcal{T}}$ be two eigenvectors resulting from diffusion maps applied (separately) to $\{\boldsymbol{y}_v\}$ and $\{\boldsymbol{y}_t\}$, respectively. Then, an alternative definition of the filters in Eq. **12** can be

$$g_{\ell,\ell'} = \boldsymbol{\psi}_\ell^{\mathcal{V}} \otimes \boldsymbol{\psi}_{\ell'}^{\mathcal{T}}. \quad [17]$$

These two variants will be further explored in future work.

Appendix 3: More on the Two Coupled Pendula

Time-Varying Spring Constant. Our two coupled pendula illustration was chosen so as to provide an easily verifiable ground truth example in the linearized regime and do so involving not only static but also time-dependent inputs. However, even in this linearized regime, when the spring constant varies in time, the system is not solvable in closed form. The evolution equations read

$$\begin{cases} m\ddot{u}^{(1)} = -\frac{mg}{L}u^{(1)} - k(t)\left(u^{(2)} - u^{(1)}\right) \\ m\ddot{u}^{(2)} = -\frac{mg}{L}u^{(2)} + k(t)\left(u^{(2)} - u^{(1)}\right). \end{cases} \quad [18]$$

In the typical constant k scenario, the system is described by two "normal modes," which are obtained by rotating the coordinate system by $45°$, leading to two independent ODEs:

$$\begin{cases} \ddot{v}^{(1)} = -\omega_1^2 v^{(1)} \\ \ddot{v}^{(2)} = -\omega_2^2(t) v^{(2)}, \end{cases} \quad [19]$$

where $v^{(1)} \triangleq u^{(1)} + u^{(2)}$, $v^{(2)} \triangleq u^{(1)} - u^{(2)}$, and

$$\omega_1 = \sqrt{\frac{g}{L}}, \quad \omega_2(t) = \sqrt{\frac{g}{L} + \frac{2k(t)}{m}}.$$

The solution of the first differential equation establishes the first normal mode: a solution with a single oscillation frequency $v^{(1)}(t) = A_1 \cos(\omega_1 t)$, where A_1 is a constant depending on the initial conditions. The second equation is of the form of a Schrödinger equation with time-varying potential, $\ddot{v}^{(2)} + \omega_2^2(t) v^{(2)} = 0$, and there is no known analytical solution method in the current literature. However, several numerical methods exist (e.g., ref. 48).

If we assume a constant spring, then the second equation becomes linear and similar to the first. Specifically, for the following initial conditions,

$$\dot{u}^{(1)}(0) = 0, u^{(1)}(0) = \delta, \dot{u}^{(2)}(0) = 0, u^{(2)}(0) = 0,$$

where $\dot{u}$ is the first derivative of u, and $0 < \delta \in \mathbb{R}$ is assumed to be sufficiently small to make the linearization an acceptable approximation; the closed form solution of the ODE in Eq. **7** consists of a linear combination of two normal modes (oscillations) and is given by

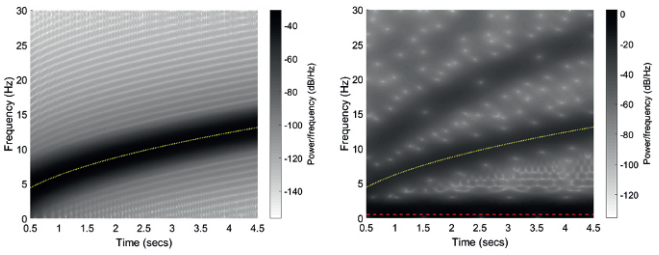

Fig. 7. (*Left*) The Fourier spectrogram of the numerical solution of $v^{(2)}$ in Eq. **19**. (*Right*) The Fourier spectrogram of the principal eigenvector representing the time axis. The two frequencies ω_1 and $\omega_2(t)$ are overlaid on the spectrograms. The dashed red line corresponds to the fixed oscillation frequency ω_1, and the dotted yellow lines correspond to the time-varying oscillation frequency $\omega_2(t)$.

$$\begin{cases} u^{(1)}(t) = \frac{1}{2}\delta\cos(\omega_1 t) + \frac{1}{2}\delta\cos(\omega_2 t) \\ u^{(2)}(t) = \frac{1}{2}\delta\cos(\omega_1 t) - \frac{1}{2}\delta\cos(\omega_2 t) \end{cases}, \quad [20]$$

where

$$\omega_1 = \sqrt{\frac{g}{L}},\ \omega_2 = \sqrt{\frac{g}{L} + \frac{2k}{m}}.$$

An Additional Experiment. We repeat the experiment reported in *Two Coupled Pendula*. Here, to help bridge the gap between the two experiment variants (one with a fixed spring constant and verifiable ground truth and the other with time-varying spring constant without known definitive analytic foundation), we simulate a system with a particular time-varying spring constant, so that the system has known numerical solution. The simulated time-varying spring constant used here is $k(t) = 1{,}500t + 10$. The remaining details of the simulation are left unchanged and described in the text.

The empirical solution of $v^{(2)}$ in Eq. **19** is displayed in Fig. 7, *Left*. The results of the application of *Algorithm 1* are presented in Fig. 7, *Right*, where the Fourier spectrogram of the principal eigenvector is shown. Here, as well, we observe that the same two frequencies ω_1 and $\omega_2(t)$ are captured by our method, despite the additional unknown observation function. In addition, the results imply that, indeed, our method empirically recovers an accurate solution even in this "more complex" scenario with time-varying spring constant. Moreover, they suggest that the empirical solution obtained by our method may be reliable even in cases for which we have no known ground truth, such as the case described in the text. In light of these findings, we believe that our approach can serve as an important, viable, empirical tool in the investigation of nonlinear dynamical systems with time-varying parameters.

Algorithm 1. *Iterative analysis of dynamical systems data.*

***Input*:** 3D data tensor $\mathbf{Y}$, initial metrics on samples $\boldsymbol{y}_v$ from the variables axis and on samples $\boldsymbol{y}_t$ from the time axis [i.e., $\|\cdot\|_{\mathcal{V}}^{(0)}$ and $\|\cdot\|_{\mathcal{T}}^{(0)}$].

***Construction*:** Set $n = 0$.

***Repeat*:**

i) Call *Algorithm 2* with the metrics $\|\cdot\|_{\mathcal{V}}^{(n)}$ and $\|\cdot\|_{\mathcal{T}}^{(n)}$ as inputs, and obtain the informed metric $\|\boldsymbol{y}_p - \boldsymbol{y}_l\|_{\mathcal{P}}^{(n+1)}$.

ii) Apply diffusion maps with the metric $\|\boldsymbol{y}_p - \boldsymbol{y}_l\|_{\mathcal{P}}^{(n+1)}$, and obtain the nth iteration embedding of samples $\boldsymbol{y}_p$ from the parameters axis.

iii) Call *Algorithm 2* with the metrics $\|\cdot\|_{\mathcal{P}}^{(n+1)}$ and $\|\cdot\|_{\mathcal{T}}^{(n)}$ as inputs, and obtain the informed metric $\|\boldsymbol{y}_v - \boldsymbol{y}_\nu\|_{\mathcal{V}}^{(n+1)}$.

iv) Apply diffusion maps with the metric $\|\boldsymbol{y}_v - \boldsymbol{y}_\nu\|_{\mathcal{V}}^{(n+1)}$, and obtain the nth iteration embedding of the samples $\boldsymbol{y}_v$ from the variables axis.

v) Call *Algorithm 2* with the metrics $\|\cdot\|_{\mathcal{P}}^{(n+1)}$ and $\|\cdot\|_{\mathcal{V}}^{(n+1)}$ as inputs, and obtain the informed metric $\|\boldsymbol{y}_t - \boldsymbol{y}_\tau\|_{\mathcal{T}}^{(n+1)}$.

vi) Apply diffusion maps with the metric $\|\boldsymbol{y}_t - \boldsymbol{y}_\tau\|_{\mathcal{T}}^{(n+1)}$, and obtain the nth iteration embedding of samples $\boldsymbol{y}_t$ from the time axis.

vii) Set $n = n + 1$, and jump to step *i*.

Algorithm 2. *Informed metric computation based on trees.*

***Input*:** 3D data tensor $\mathbf{Y}$, metrics on two axes: $\|\cdot\|_{(a)}, \|\cdot\|_{(b)}$.

***Output*:** A metric on the third axis $\|\cdot\|_{(c)}$.

***Initialization*:**

i) Construct a partition tree $\mathcal{T}_{(a)}$ of the samples $\boldsymbol{y}_a$ based on the metric $\|\cdot\|_{(a)}$.

ii) Construct a partition tree $\mathcal{T}_{(b)}$ of the samples $\boldsymbol{y}_b$ based on the metric $\|\cdot\|_{(b)}$.

***Construction*:**

i) Build the filters $g_{I\times J}$ for all $I \in \mathcal{T}_{(a)}$ and $J \in \mathcal{T}_{(b)}$ as in Eq. **12**.

ii) Transform the samples $\boldsymbol{y}_c$ by computing $\mathcal{F}_{(c)}(\boldsymbol{y}_c)$ as in Eq. **16**.

iii) Compute the informed metric

$$\|\boldsymbol{y}_c - \boldsymbol{y}_{c'}\|_{(c)} = \|\boldsymbol{y}_c - \boldsymbol{y}_{c'}\|_1 + \gamma\|\mathcal{F}_{(c)}(\boldsymbol{y}_c) - \mathcal{F}_{(c)}(\boldsymbol{y}_{c'})\|_1$$

between all possible pairs of samples.

ACKNOWLEDGMENTS. The work of O.Y. and R.T. was supported by the European Union's Seventh Framework Program (FP7) under Marie Curie Grant 630657 and Israel Science Foundation Grant 1490/16. The work of R.T. and R.R.C. was supported by National Science Foundation Grant 1309858. R.T. and I.G.K. acknowledge the support and hospitality of Institute for Advanced Study–Technical University of Munich. The work of I.G.K. was supported by the National Science Foundation, the Air Force Office of Scientific Research (Dr. Darema), and the Defense Advanced Research Projects Agency Contract HR0011-16-C-0016.

1. Bongard J, Lipson H (2007) Automated reverse engineering of nonlinear dynamical systems. *Proc Natl Acad Sci USA* 104:9943–9948.
2. Schmidt M, Lipson H (2009) Distilling free-form natural laws from experimental data. *Science* 324:81–85.
3. Crutchfield JP, McNamara BS (1987) Equations of motion from a data series. *Complex Syst* 1:417–452.
4. Doretto G, Chiuso A, Wu YN, Soatto S (2003) Dynamic textures. *Int J Computer Vis* 51:91–109.
5. Brunton SL, Proctor JL, Kutz JN (2016) Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc Natl Acad Sci USA* 113:3932–3937.
6. Kevrekidis IG, et al. (2003) Equation-free, coarse-grained multiscale computation enabling mocroscopic simulators to perform system-level analysis. *Commun Math Sci* 1:715–762.
7. Sugihara G, et al. (2012) Detecting causality in complex ecosystems. *Science* 338: 496–500.
8. Tenenbaum JB, de Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. *Science* 290:2319–2323.
9. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. *Science* 290:2323–2326.
10. Donoho DL, Grimes C (2003) Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data. *Proc Natl Acad Sci USA* 100:5591–5596.
11. Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput* 15:1373–1396.
12. Coifman RR, Lafon S (2006) Diffusion maps. *Appl Comput Harmon Anal* 21:5–30.
13. Gavish M, Coifman RR (2012) Sampling, denoising and compression of matrices by coherent matrix organization. *Appl Comput Harmon Anal* 33:354–369.
14. Ankenman JI (2014) Geometry and analysis of dual networks on questionnaires. PhD thesis (Yale University, New Haven, CT).
15. Mishne G, et al. (2015) Hierarchical coupled geometry analysis for neuronal structure and activity pattern discovery. arXiv:1511.02086.
16. Leeb WE (2015) Topics in metric approximation. PhD thesis (Yale University, New Haven, CT).
17. Budišić M, Mezić I (2012) Geometry of the ergodic quotient reveals coherent structures in flows. *Physica D* 241:1255–1269.
18. Mezic I (2016) On comparison of dynamics of dissipative and finite-time systems using koopman operator methods. *IFAC-PapersOnLine* 49:454–461.
19. Mezić I, Banaszuk A (2004) Comparison of systems with complex behavior. *Physica D* 197:101–133.
20. Guckenheimer J, Holmes PJ (2013) *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields* (Springer, New York), Vol 42.
21. Lafon SS (2004) Diffusion maps and geometric harmonics. PhD thesis (Yale University, New Haven, CT).
22. Singer A, Coifman RR (2008) Non-linear independent component analysis with diffusion maps. *Appl Comput Harmon Anal* 25:226–239.
23. Giannakis D (2015) Dynamics-adapted cone kernels. *SIAM J Appl Dyn Syst* 14: 556–608.
24. Dsilva CJ, Talmon R, Gear CW, Coifman RR, Kevrekidis IG (2016) Data-driven reduction for a class of multiscale fast-slow stochastic dynamical systems. *SIAM J Appl Dyn Syst* 15:1327–1351.
25. Coifman RR, Gavish M (2011) Harmonic analysis of digital data bases. *Wavelets and Multiscale Analysis*, Applied and Numerical Harmonic Analysis, eds Cohen J, Zayed AI (Birkhäuser, Boston), pp 161–197.

26. Rubner Y, Tomasi C, Guibas LJ (2000) The earth mover's distance as a metric for image retrieval. *Int J Comput Vis* 40:99–121.
27. Rubner Y, Tomasi C (2001) The earth mover's distance. *Perceptual Metrics for Image Database Navigation* (Springer, Berlin), pp 13–28.
28. Shirdhonkar S, Jacobs DW (2008) Approximate earth movers distance in linear time. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ), pp 1–8.
29. Coifman R, Leeb W (2013) Earth mover's distance and equivalent metrics for spaces with hierarchical partition trees (Yale University, New Haven, CT), Technical Report YALEU/DCS/TR-1482.
30. Comon P, Luciani X, De Almeida AL (2009) Tensor decompositions, alternating least squares and other tales. *J Chemom* 23:393–405.
31. Chi EC, Allen GI, Baraniuk RG (2016) Convex biclustering. *Biometrics* 73:10–19.
32. Anandkumar A, Ge R, Hsu D, Kakade SM, Telgarsky M (2014) Tensor decompositions for learning latent variable models. *J Mach Learn Res* 15:2773–2832.
33. Kalman RE (1963) Mathematical description of linear dynamical systems. *J Soc Ind Appl Math Ser A: Contr* 1:152–192.
34. Moore B (1981) Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans Automatic Contr* 26:17–32.
35. Lall S, Marsden JE, Glavaški S (1999) Empirical model reduction of controlled nonlinear systems. *Proceedings of the IFAC World Congress* (International Federation of Automatic Control, New York), pp 473–478.
36. Packard NH, Crutchfield JP, Farmer JD, Shaw RS (1980) Geometry from a time series. *Phys Rev Lett* 45:712–716.
37. Whitney H (1944) The self-intersections of a smooth n-manifold in 2n-space. *Ann Math* 45:220–246.
38. Takens F (1981) Detecting strange attractors in turbulence. *Dynamical Systems and Turbulence, Warwick 1980* (Springer, Berlin), pp 366–381.
39. Nash J (1956) The imbedding problem for riemannian manifolds. *Ann Math* 63: 20–63.
40. Talmon R, Coifman RR (2013) Empirical intrinsic geometry for nonlinear modeling and time series filtering. *Proc Natl Acad Sci USA* 110:12535–12540.
41. Dsilva CJ, Talmon R, Rabin N, Coifman RR, Kevrekidis IG (2013) Nonlinear intrinsic variables and state reconstruction in multiscale simulations. *J Chem Phys* 139:184109.
42. Talmon R, Coifman RR (2015) Intrinsic modeling of stochastic dynamical systems using empirical geometry. *Appl Comput Harmon Anal* 39:138–160.
43. Dsilva CJ, et al. (2015) Temporal ordering and registration of images in studies of developmental dynamics. *Development* 142:1717–1724.
44. Lederman RR, Talmon R (2015) Learning the geometry of common latent variables using alternating-diffusion. *Appl Comput Harmon Anal*, 10.1016/j.acha.2015.09.002.
45. Breiman L, Friedman JH (1985) Estimating optimal transformations for multiple regression and correlation. *J Am Stat Assoc* 80:580–598.
46. Hein M, Audibert JY (2005) Intrinsic dimensionality estimation of submanifolds in r d. *Proceedings of the 22nd International Conference on Machine Learning* (ACM), pp 289–296.
47. Coifman RR, Shkolnisky Y, Sigworth FJ, Singer A (2008) Graph laplacian tomography from unknown random projections. *IEEE Trans Image Process* 17:1891–1899.
48. Bremer J, Rokhlin V (2015) Improved estimates for nonoscillatory phase functions. arXiv:1505.05548.