Editors' Suggestion

Machine Learning Conservation Laws from Trajectories

Ziming Liu* and Max Tegmark
Department of Physics, Institute for AI and Fundamental Interactions, and Center for Brains, Minds and Machines, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

(Received 9 November 2020; revised 20 January 2021; accepted 15 April 2021; published 6 May 2021)

We present AI Poincaré, a machine learning algorithm for autodiscovering conserved quantities using trajectory data from unknown dynamical systems. We test it on five Hamiltonian systems, including the gravitational three-body problem, and find that it discovers not only all exactly conserved quantities, but also periodic orbits, phase transitions, and breakdown timescales for approximate conservation laws.

DOI: 10.1103/PhysRevLett.126.180604

Introduction.—While machine learning has contributed to many physics advances, such as improving the speed or quality of numerical simulations, laboratory experiments and astronomical observations [1–7], a more ambitious goal is to design intelligent machines to make new scientific discoveries such as physical symmetries [8–12] and formulas via symbolic regression [13–17]. In this spirit, the goal of the present Letter is to autodiscover conservation laws from trajectories of dynamical systems.

Physicists have traditionally derived conservation laws in a *model-driven* way, such as when Poincaré proved [18] that the 3D gravitational three-body problem has only ten conserved quantities. In contrast, this Letter aims to discover conservation laws in a *data-driven* way, using only observed trajectory data as input while treating the underlying dynamical equations as unknown.

To the best of our knowledge, the authors of [11,12] have pursued the goal closest to ours, but with an orthogonal approach detecting symmetry with an autoencoder and Siamese neural networks, respectively, requiring handcrafted features precluding full automation and testing on relatively simple examples. Other work linking conservation laws and machine learning [9,19–21] focus on embedding physical inductive biases (such as the existence of a Hamiltonian or Lagrangian) into machine learning, but not the other way around to apply machine learning for autodiscovery of conservation laws.

Our ambitious goal of automating conservation law discovery is enabled by recent machine learning progress [22] for sampling *manifolds*, which are intimately related to dynamical systems as summarized in Table I: viewing each state as a point in a phase space $\mathbb{R}^N$, the topological closure of the set of all states on a trajectory form a manifold $\mathcal{M} \subset \mathbb{R}^N$. Each conservation law removes one degree of freedom from the dynamical system and one dimension from $\mathcal{M}$, so the number of conserved quantities is simply N minus the dimensionality of $\mathcal{M}$ [23]. The local tangent space of $\mathcal{M}$ represents all local displacements allowed by conservation laws, while the space perpendicular to the tangent space is spanned by gradients of conserved quantities.

We introduce our notation and AI Poincaré algorithm in the methods section. In the results section, we apply AI Poincaré to five Hamiltonian systems to test its ability to discover conserved quantities (numerically and symbolically), periodic orbits, phase transitions, and conservation breakdown timescales.

Method.—Problem and notation: Consider a dynamical system whose state vector $\mathbf{x} \in \mathbb{R}^N$ evolves according to an ordinary differential equation $d\mathbf{x}/dt = f(\mathbf{x}, t)$ for some smooth function f. In physics, dynamical systems can often be written with $\mathbf{x}$ as the concatenation of vectors of generalized coordinates $\mathbf{q}$ and momenta $\mathbf{p}$ and N is even. For the special case of *Hamiltonian* systems, important in classical mechanics [23], there exists a Hamiltonian function $H_0(\mathbf{q}, \mathbf{p})$ such that

$$\frac{d\mathbf{p}^{(i)}}{dt} = -\frac{\partial H_0}{\partial \mathbf{q}^{(i)}}, \qquad \frac{d\mathbf{q}^{(i)}}{dt} = \frac{\partial H_0}{\partial \mathbf{p}^{(i)}} \quad (i = 1, \ldots, k). \tag{1}$$

Conservation laws are important in physics, common examples including conservation of energy (H_0), momentum, angular momentum, Runge-Lenz vector, magnetic moment, etc. We express a set of independent conservation laws as $H_j[\mathbf{x}(t)] = h_j$, $j = 0, \ldots, n-1$, valid exactly or approximately. Each conservation law can be understood as a mathematical constraint that slices the original n-dimensional phase space into a family of isosurfaces.

TABLE I. Manifold–dynamical system correspondence.

Manifold	Dynamical system
Dimensionality reduction	Conservation law
Tangent space	Conserved quantity isosurface
Orthogonal space	Gradients of conserved quantities

We define the permissible state manifold (PSM) as $\mathcal{M} \equiv \{\mathbf{x} \in \mathbb{R}^N | H_j(\mathbf{x}) = h_j\}$, i.e., as the set of states allowed by all conservation laws. It is clear that $\dim(\mathcal{M}) = N - n$ [23], since each conservation law removes one degree of freedom from the system. In practice, however, only trajectory data rather than the full PSM is observed, which motivates us to define the "trajectory set" as $\mathcal{S} = \{\mathbf{x}(t) | t \geq 0\}$. If the system is ergodic, $\mathcal{S}$ will be everywhere dense in $\mathcal{M}$ so that $\mathcal{M}$ is the closure $\bar{\mathcal{S}}$ of $\mathcal{S}$, implying the identity $n = N - \dim(\bar{\mathcal{S}})$. This generalizes even beyond traditional physics contexts: for example, if $\mathbf{x}$ contains the pixel colors in a gray-scale video, then the color of a pixel that always remains black is a conserved quantity.

AI Poincaré: We present a machine learning algorithm, AI Poincaré, that uses S to compute an estimator $\hat{s}$ of the dimensionality $s \equiv \dim(\bar{\mathcal{S}})$, thus obtaining an estimator $n_{\text{eff}} \equiv N - \hat{s}$ of n. If $n_{\text{eff}} > 0$, it suggests the existence of n_{eff} hitherto undiscovered conservation laws. If $n_{\text{eff}} = 0$, the system is not Hamiltonian since it lacks even a single conserved quantity H_0.

Although manifold learning is an active field and has developed powerful tools to explore and visualize the latent structure of low-dimensional submanifolds of high-dimensional spaces [24–29], they either focus on performance on downstream tasks (e.g., image or video generation), where the dimensionality is a user-specified input parameter, or perform not as well as the proposed method here on our task. The Supplemental Material [30] compares the performance with the principal component analysis (PCA), autoencoder, and fractal methods. If we had perfect noiseless samples forming a dense set $\mathcal{S}$ in $\mathcal{M}$, then we could simply determine the manifold dimensionality s as the rank (number of nonzero eigenvalues) of the covariance matrix of the samples in an infinitesimal environment of a random sample, where the manifold can be approximated by a hyperplane. In practice, we cannot probe infinitesimal scales because we have only a finite number of points, yet we must avoid large scales where manifold curvature is important; our method handles these complications by treating dimensionality s as a renormalized quantity that is a function of length scale L.

AI Poincaré consists of three modules, as illustrated in Fig. 1(a): (1) preprocessing (prewhitening and optional dimensionality reduction), (2) local Monte Carlo sampling of $\mathcal{M}$, and (3) linear dimensionality estimation from these samples using PCA explained ratios. The prewhitening performs an affine transformation such that the points in $\mathcal{S}$ have zero mean and covariance matrix $\mathbf{I}$, the identity matrix. If any eigenvalues of the original covariance matrix vanish, then the corresponding eigenvectors $\mathbf{e}_i$ define linear conserved quantities $H_i(\mathbf{x}) = \mathbf{e}_i \cdot \mathbf{x}$, and we remove these dimensions before proceeding. The Supplemental Material [30] gives further technical details of the preprocessing module.

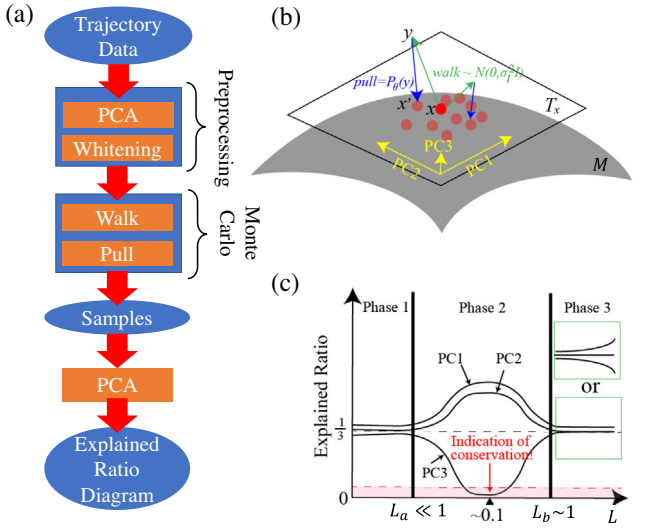


FIG. 1. The AI Poincaré algorithm: (a) overall work flow, (b) walk-pull Monte Carlo module, (c) typical explained ratio diagram, with phase 2 revealing conserved quantities. PC denotes principal component.

Our module for Monte Carlo sampling the manifold benefits from the neural empirical Bayes technique in the machine learning literature [22]. It consists of two steps, as illustrated in Fig. 1(b): (1) The walk step $\mathbf{x} \mapsto \mathbf{y}$ perturbs a state vector $\mathbf{x} \in \mathcal{S}$ by adding isotropic Gaussian noise $\mathbf{n}$ with zero mean and covariance $L\mathbf{I}$. (2) The pull step $\mathbf{y} \mapsto \mathbf{x}'$ pulls the "noisy" state $\mathbf{y}$ back toward the manifold. We do this by training the parameters $\boldsymbol{\theta}$ of a feed forward neural network implementing a "pull function" $P_{\boldsymbol{\theta}}$ mapping $\mathbb{R}^N$ to $\mathbb{R}^N$, minimizing the loss function

$$\text{Loss}(\boldsymbol{\theta}) = \frac{1}{N_s} \sum_{i=1}^{N_s} |P_{\boldsymbol{\theta}}(\mathbf{y}_i) - \mathbf{x}_i|_2^2, \tag{2}$$

where N_s is the number of samples. Since the best the neural network can do is learn to orthogonally project points back onto the manifold, the pulled-back points $\mathbf{x}'_i \equiv P_{\boldsymbol{\theta}}(\mathbf{y}_i)$ characterize the local tangent space when L is appropriately chosen [22].

The Supplemental Material [30] provides further intuition and illustrations regarding how the preprocessing walk-pull steps work.

Explained ratio diagram: The output of the AI Poincaré algorithm is the explained ratio diagram (ERD), showing the fraction of variance explained by each principal component (its eigenvalue divided by the eigenvalue sum) as a function of noise length scale L, revealing the dimensionality of $\mathcal{M}$ and hence the number of conserved quantities. A typical ERD is shown in Fig. 1(c), revealing three phases separated by phase transitions at $L_a \ll 1$ and $L_b \sim 1$. In the intermediate phase $L_a \lesssim L \lesssim L_b$, some principal component(s) drop near zero and are identified

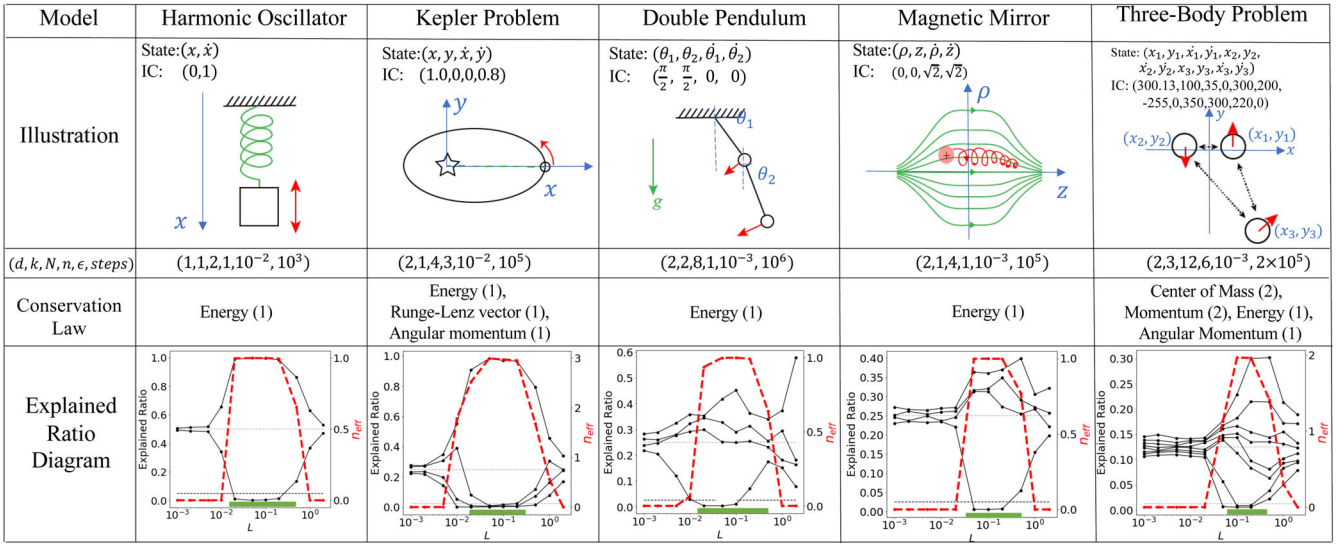


FIG. 2. Five Hamiltonian systems used to test AI Poincaré: Harmonic oscillator, Kepler problem, double pendulum, magnetic mirror, and three-body problem. d is the dimensionality of the Euclidean space, k is the number of bodies, $N \equiv 2kd$ is the phase space dimensionality, and n is the ground truth number of conservation laws. Bottom: the red dashed curve shows the effective number of conserved quantities $n_{\rm eff}$ defined by Eq. (3), and the green region on the L axis shows L range that gets $n_{\rm eff}$ correct (after rounding to the nearest integer). The horizontal dashed lines shows $1/N$ and $0.1/N$; any principal component explaining less than a fraction $0.1/N$ of the total variance at any L is considered evidence for a conservation law. IC denotes initial conditions.

as submanifold structure (conservation laws). In the other two phases, all explained ratios are $\sim 1/n$ because the Monte Carlo sample covariance matrix $\sim \mathbf{I}$: on large scales because almost the whole (prewhitened) manifold is sampled and on small scales because of roughly isotropic noise.

Results.—Numerical experiments: We test AI Poincaré on trajectories from five well-studied Hamiltonian systems: the 1D harmonic oscillator, 2D Kepler problem, double pendulum, 2D magnetic mirror, and 2D three-body problem, as defined in Table II and illustrated in Fig. 2. We compute trajectories for the five systems with a fourth-order Runge-Kutta integrator at $N_{\rm step} = \{10^3, 10^5, 10^6, 10^5, 2\times10^5\}$ time steps of size $\epsilon = \{10^{-2}, 10^{-2}, 10^{-3}, 10^{-3}, 10^{-3}\}$, using the initial conditions listed in Fig. 2. We parametrize the pull function $P_{\boldsymbol{\theta}}$ as a feed forward neural network with two hidden layers containing 256 neurons each and train it for each L using the Adam optimizer [31] with learning rate 10^{-3} for 5000 steps. We repeat the walk + pull Monte Carlo process jumping 10^4 times from the trajectory midpoint.

Basic results: The resulting explained ratios (Fig. 2, bottom) show a consistent valley around $L = 0.1$, revealing the number of conserved quantities. The number of conservation laws discovered by AI Poincaré is seen to agree with the ground truth (Table II) for all five systems if we simply define the criterion for conservation law discovery as an explained ratio that is an order of magnitude below baseline ($0.1/N$; dashed black line in the figure). For the three-body problem, the first four conservation laws are linear and hence discovered already in our preprocessing step. As shown in the Supplemental Material [30], these results are robust to changing the starting point for the walk + pull process, and outperform the PCA, autoencoder, and fractal methods for dimensionality estimation.

Symbolic formula discovery: Table II (right column) shows that we can not only autodiscover that a conservation law exists, but in many cases also a symbolic formula for it: we did this by applying the AI Feynman symbolic regression algorithm [16,17] to our trajectory data. Since any function of a conserved quantity is also conserved, we require some form of "gauge fixing" to make the symbolic regression problem well posed. Here we simply require states on two chosen trajectories to have conserved value 1 and 2; this approach can undoubtedly be greatly improved, e.g., by requiring that gradient directions match those of our pull function.

Phase transition discovery: We will now explore how AI Poincaré can autodiscover not only *exact* conservation laws as above, but also *approximate* ones, revealing physically interesting phase transitions. There are many reasonable ways of defining an effective number of conserved quantities $n_{\rm eff}$ as a smooth function of the explained ratios w_i. We want each explained ratio w_i to contribute 1 for small w_i and 0 for $w_i \gtrsim 1/N$, so here we make the simple choice

$$n_{\rm eff} \equiv \max_L n_{\rm eff}(L), \qquad n_{\rm eff}(L) \equiv \sum_{i=1}^{N} c[\pi N w_i(L)], \tag{3}$$

where $c(x) \equiv \cos x$ if $x < \pi/2$, vanishing otherwise. This is seen to agree well with our threshold criterion

TABLE II. Symbolic formulas for 10 of the 13 conservation laws that were discovered using AI Feynman.

System	Conserved quantity	Formula found?
Harmonic oscillator	$H = \frac{1}{2}(x^2 + \dot{x}^2)$	Yes
Kepler problem	$H = \frac{1}{2}(\dot{x}^2 + \dot{y}^2) - \left(1/\sqrt{x^2 + y^2}\right)$	Yes
	$L_{\text{angular}} = x\dot{y} - y\dot{x}$	Yes
	$A = \arg[L(-\dot{y}, \dot{x}) - \hat{r}]$	No
Double pendulum	(Small angle) $H_s = 10\theta_1^2 + 5\theta_2^2 + \dot{\theta}_1{}^2 + \frac{1}{2}\dot{\theta}_2{}^2 + \dot{\theta}_1\dot{\theta}_2$	Yes
	(Large angle) $H_l = -20\cos\theta_1 - 10\cos\theta_2 + \dot{\theta}_1{}^2 + \frac{1}{2}\dot{\theta}_2{}^2 + \dot{\theta}_1\dot{\theta}_2\cos(\theta_1 - \theta_2)$	No
Magnetic mirror	$H = \frac{1}{2}(\dot{\rho}^2 + \dot{z}^2) + \frac{1}{2}(\rho^2 + \frac{1}{5}z^2 + \rho^2 z^2)$	Yes
Three-body problem	$H = \sum_{i=1}^{3}\frac{m}{2}(\dot{x}_i^2 + \dot{y}_i^2) - m^2(\frac{1}{r_{12}} + \frac{1}{r_{13}} + \frac{1}{r_{23}}), m = 5 \times 10^6$	No
	$x_c = \frac{1}{3}(x_1 + x_2 + x_3)$	Yes
	$y_c = \frac{1}{3}(y_1 + y_2 + y_3)$	Yes
	$\dot{x}_c = \frac{1}{3}(\dot{x}_1 + \dot{x}_2 + \dot{x}_3)$	Yes
	$\dot{y}_c = \frac{1}{3}(\dot{y}_1 + \dot{y}_2 + \dot{y}_3)$	Yes
	$L_{\text{angular}} = \sum_{i=1}^{3} x_i\dot{y}_i - y_i\dot{x}_i$	Yes

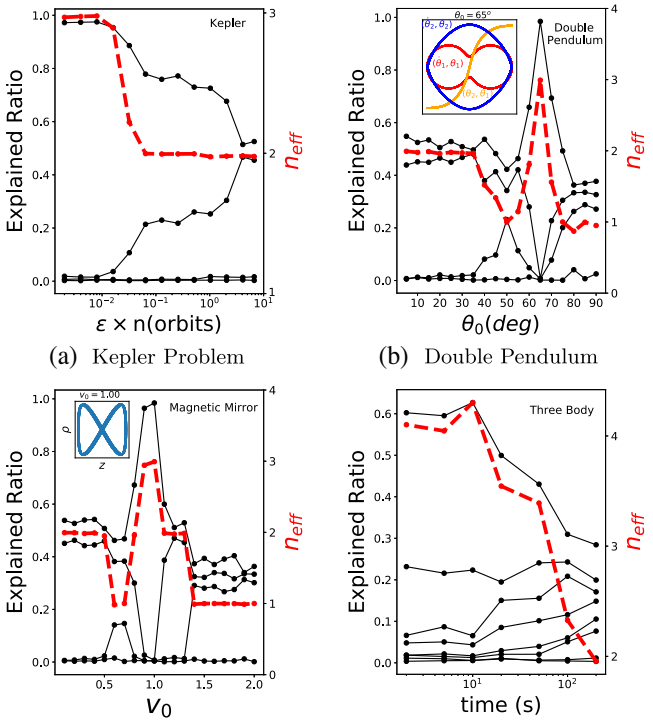
(Fig. 2, bottom). Below we fix $L = 0.1$, instead of maximizing over it to save computation time. Let us view n_{eff} as an "order parameter" of the dynamical system and consider phase transitions in the parameter space spanned by timescale, initial conditions, and Hamiltonian modifications.

For the Kepler problem, we generalize the inverse square force to the form $F \propto r^{-(2+\epsilon)}$, $|\epsilon| \ll 1$. This causes the previously conserved Runge-Lenz vector (the major axis direction) to precess by an angle $\sim\epsilon$ per orbit [32], so that its approximate conservation breaks down after $\sim\epsilon^{-1}$ orbits. AI Poincaré is seen to autodiscover this phase transition [Fig. 3(a)] without using any of the aforementioned physics knowledge.

The double pendulum is known to have a regular phase at low energy and a chaotic phase at high energy [33], both with $n_{\text{eff}} = 1$ (conserving only total energy). We change the initial conditions to $\theta_1 = \theta_2 = \theta_0$ and plot the dependence of n_{eff} on θ_0 [Fig. 3(b)], which is seen to reveal two additional phases. (1) An interesting periodic orbit ($n_{\text{eff}} = 3$) is discovered at $\theta_0 \approx 65°$ (see inset figure), by adjusting initial conditions to maximize n_{eff}. (2) For small θ_0, the small angle approximation allows the system to be accurately linearized and decoupled into two noninteracting normal modes, whose energies are separately conserved ($n_{\text{eff}} = 2$).

The magnetic mirror is also known to have a regular low-energy phase and a chaotic high-energy phase [34], both with $n_{\text{eff}} = 1$ (conserving only energy). We change the initial conditions to $\dot{\rho} = \dot{z} = v_0/\sqrt{2}$ and the dependence of n_{eff} on v_0 in Fig. 3(c) is seen to reveal two additional phases. (1) An interesting periodic orbit ($n_{\text{eff}} = 3$) is discovered at $v_0 \approx 1.0$ (see inset figure). (2) At low energy (small v_0), the magnetic moment is an adiabatic invariant, so $n_{\text{eff}} = 2$.

For the three-body problem, we consider initial conditions akin to a tight binary star pair orbiting a more distant star. Figure 3(d) reveals that the local energy and angular momentum of the tight binary are approximately conserved



(a) Kepler Problem (b) Double Pendulum (c) Magnetic Mirror (d) Three-body Problem

FIG. 3. AI Poincaré detection of phase transitions, approximate conserved quantities and periodic orbits, using n_{eff} as an order parameter. Four examples were tested: (a) Kepler problem, (b) double pendulum, (c) magnetic mirror, and (d) three-body problem.

initially, increasing n_{eff} by two, until tidal interactions with the distant star eventually cause this conservation to break down.

Conclusions.—We have presented AI Poincaré, a machine learning algorithm for autodiscovering conserved quantities using trajectory data from unknown dynamical systems. Tests on five Hamiltonian systems showed that it autodiscovered not only the number of conserved quantities, but also periodic orbits, phase transitions, and breakdown timescales for approximate conservation laws. AI Poincaré is universal in the sense that it does not require any domain knowledge or even a physical model of how the trajectories were produced. It may therefore be interesting to apply it to raw experimental data, for example, measured neuron voltages in *C. elegans*. Another promising future direction is improved discovery of symbolic formulas for the discovered conserved quantities by transferring learned geometric information from AI Poincaré to AI Feynman, e.g., by requiring that symbolic gradient directions match those of our learned pull function.

We thank Qihao Cheng, Artan Sheshmani, and Huichao Song for helpful discussions and the Center for Brains, Minds, and Machines (CBMM) for hospitality. This work was supported by the Casey and Family Foundation, the Foundational Questions Institute, the Rothberg Family Fund for Cognitive Science, and Institute for Artificial Intelligence and Fundamental Interactions (IAIFI) through NSF Grant No. PHY-2019786.

*zmliu@mit.edu

[1] E. van Nieuwenburg, Y.-H. Liu, and S. Huber, Learning phase transitions by confusion, Nat. Phys. **13**, 435 (2017).
[2] Y. D. Hezaveh, L. P. Levasseur, and P. J. Marshall, Fast automated analysis of strong gravitational lenses with convolutional neural networks, Nature (London) **548**, 555 (2017).
[3] N. Sun, J. Yi, P. Zhang, H. Shen, and H. Zhai, Deep learning topological invariants of band insulators, Phys. Rev. B **98**, 085402 (2018).
[4] P. Baldi, P. Sadowski, and D. Whiteson, Searching for exotic particles in high-energy physics with deep learning, Nat. Commun. **5**, 4308 (2014).
[5] L.-G. Pang, K. Zhou, N. Su, H. Petersen, H. Stöcker, and X.-N. Wang, An equation-of-state-meter of quantum chromodynamics transition from deep learning, Nat. Commun. **9**, 210 (2018).
[6] M. Raissi, P. Perdikaris, and G. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. **378**, 686 (2019).
[7] M. S. Albergo, G. Kanwar, and P. E. Shanahan, Flow-based generative models for Markov chain Monte Carlo in lattice field theory, Phys. Rev. D **100**, 034515 (2019).
[8] A. Decelle, V. Martin-Mayor, and B. Seoane, Learning a local symmetry with neural networks, Phys. Rev. E **100**, 050102(R) (2019).
[9] M. Mattheakis, P. Protopapas, D. Sondak, M. D. Giovanni, and E. Kaxiras, Physical symmetries embedded in neural networks, arXiv:1904.08991.
[10] R. Bondesan and A. Lamacraft, Learning symmetries of classical integrable systems, in *ICML 2019 Workshop on Theoretical Physics for Deep Learning*, arXiv:1906.04645.
[11] Y. ichi Mototake, Interpretable conservation law estimation by deriving the symmetries of dynamics from trained deep neural networks, Phys. Rev. E **103**, 033303 (2021).
[12] S. J. Wetzel, R. G. Melko, J. Scott, M. Panju, and V. Ganesh, Discovering symmetry invariants and conserved quantities by interpreting Siamese neural networks, Phys. Rev. Research **2**, 033499 (2020).
[13] S. Kim, P. Y. Lu, S. Mukherjee, M. Gilbert, L. Jing, V. eperi, and M. Soljai, Integration of neural network-based symbolic regression in deep learning for scientific discovery, arXiv:1912.04825.
[14] P. Y. Lu, S. Kim, and M. Soljai, Extracting Interpretable Physical Parameters from Spatiotemporal Systems Using Unsupervised Learning, Phys. Rev. X **10**, 031056 (2020).
[15] M. Cranmer, A. Sanchez-Gonzalez, P. Battaglia, R. Xu, K. Cranmer, D. Spergel, and S. Ho, Discovering symbolic models from deep learning with inductive biases, arXiv:2006.11287.
[16] S.-M. Udrescu and M. Tegmark, Ai Feynman: A physics-inspired method for symbolic regression, Sci. Adv. **6**, eaay2631 (2020).
[17] S.-M. Udrescu, A. Tan, J. Feng, O. Neto, T. Wu, and M. Tegmark, Ai Feynman 2.0: Pareto-optimal symbolic regression exploiting graph modularity, arXiv:2006.10782.
[18] H. Poincaré, Les méthodes nouvelles de la mécanique céleste, Gauthier-Villars, 1 (1892–1899).
[19] M. Cranmer, S. Greydanus, S. Hoyer, P. Battaglia, D. Spergel, and S. Ho, Lagrangian neural networks, arXiv:2003.04630.
[20] S. Greydanus, M. Dzamba, and J. Yosinski, Hamiltonian neural networks, arXiv:1906.01563.
[21] M. Lutter, C. Ritter, and J. Peters, Deep Lagrangian networks: Using physics as model prior for deep learning, arXiv:1907.04490.
[22] S. Saremi and A. Hyvärinen, Neural empirical Bayes, J. Mach. Learn. Res. **20**, 1 (2019).
[23] H. Goldstein, *Classical Mechanics* (Addison-Wesley, Reading, MA, 1980).
[24] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science **290**, 2319 (2000).
[25] S. T. Roweis and L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science **290**, 2323 (2000).
[26] L. V. D. Maaten and G. E. Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. **9**, 2579 (2008).
[27] B. Luo, R. C. Wilson, and E. R. Hancock, Spectral embedding of graphs, Pattern Recognit. **36**, 2213 (2003).
[28] D.-Y. Tzeng and R. S. Berns, A review of principal component analysis and its applications to color technology, Color Research and Application **30**, 84 (2005).

[29] M. A. Kramer, Nonlinear principal component analysis using autoassociative neural networks, AIChE J. **37**, 233 (1991).

[30] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.126.180604 for technical details.

[31] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.

[32] J. D. Wells, When effective theories predict: The inevitability of Mercury's anomalous perihelion precession, arXiv:1106.1568.

[33] T. Stachowiak and T. Okada, A numerical analysis of chaos in the double pendulum, Chaos Solitons Fractals **29**, 417 (2006).

[34] G. Contopoulos, M. Harsoula, and C. Efthymiopoulos, Analytical study of chaos and applications, Eur. Phys. J. Special Topics **225**, 1053 (2016).