

Physical Models for Data Analysis

Discussion 4

May 29, 2020

Mutual Information between the Past and Future

$$\mathcal{I}_{\text{pred}}(\mathcal{T}, \mathcal{T}') = \left\langle \log_2 \left[\frac{P(x_{\text{future}} | x_{\text{past}})}{P(x_{\text{future}})} \right] \right\rangle \quad (3.1)$$

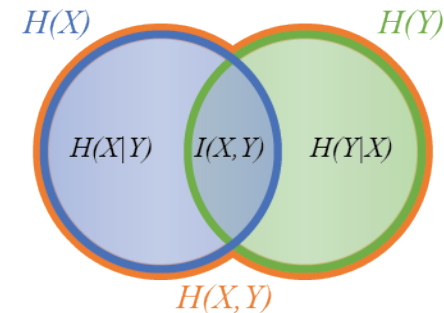
$$\begin{aligned} &= -\langle \log_2 P(x_{\text{future}}) \rangle - \langle \log_2 P(x_{\text{past}}) \rangle \\ &\quad - [-\langle \log_2 P(x_{\text{future}}, x_{\text{past}}) \rangle], \end{aligned} \quad (3.2)$$

SAME
↑
↓

$$\mathcal{I}_{\text{pred}}(\mathcal{T}, \mathcal{T}') = S(\mathcal{T}) + S(\mathcal{T}') - S(\mathcal{T} + \mathcal{T}'). \quad (3.3)$$

Mutual information

$$\begin{aligned} I(X, Y) &= H(X) - H(X | Y) \\ &= H(Y) - H(Y | X) \\ &= H(X) + H(Y) - H(X, Y) \end{aligned}$$



Sub-extensive Entropy

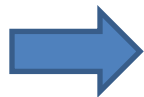
We know two general facts about the behavior of $S_1(T)$. First, the corrections to extensive behavior are positive, $S_1(T) \geq 0$. Second, the statement that entropy is extensive is the statement that the limit

$$\lim_{T \rightarrow \infty} \frac{S(T)}{T} = \mathcal{S}_0 \quad (3.4)$$

exists, and for this to be true we must also have

$$\lim_{T \rightarrow \infty} \frac{S_1(T)}{T} = 0. \quad (3.5)$$

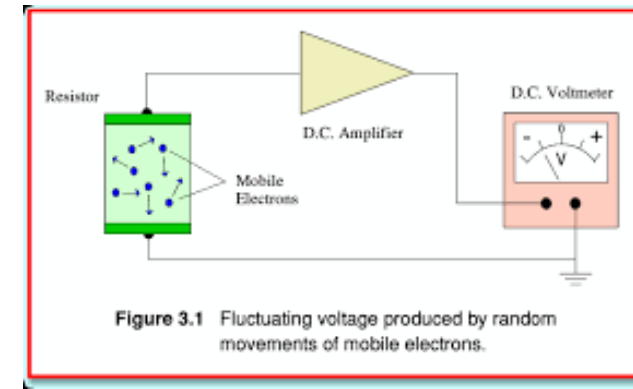
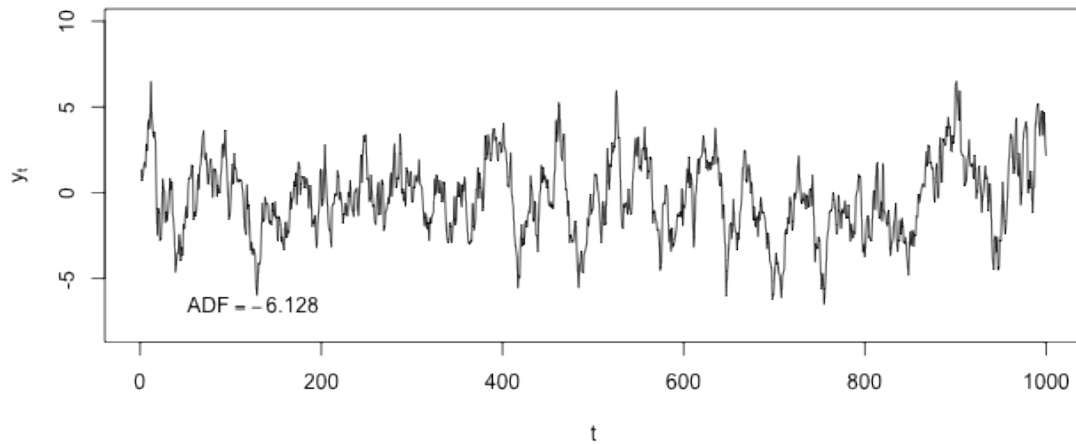
Thus, the nonextensive terms in the entropy must be *subextensive*, that is, they must grow with T less rapidly than a linear function. Taken together, these facts guarantee that the predictive information is positive and subextensive. Furthermore, if we let the future extend forward for a very long time, $T' \rightarrow \infty$, then we can measure the information that our sample provides about the entire future:



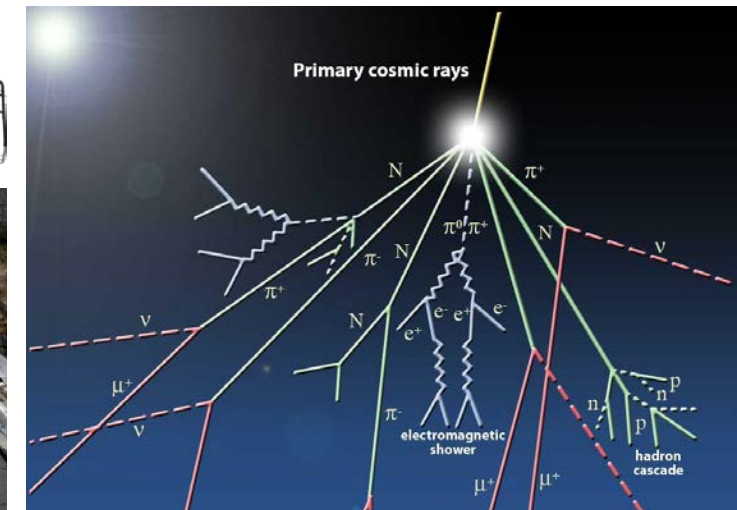
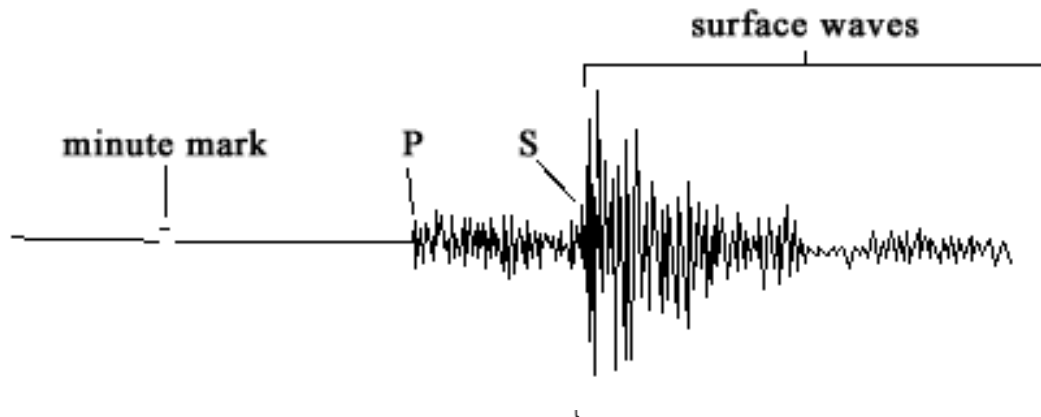
$$I_{\text{pred}}(T) = \lim_{T' \rightarrow \infty} \mathcal{I}_{\text{pred}}(T, T') = S_1(T). \quad (3.6)$$

Stationary vs. Non-stationary Time-Series

Stationary Time Series

























Non-stationary Time Series



TIME-SERIES EXERCISE

Time-series exercise on Moodle

Supporting material

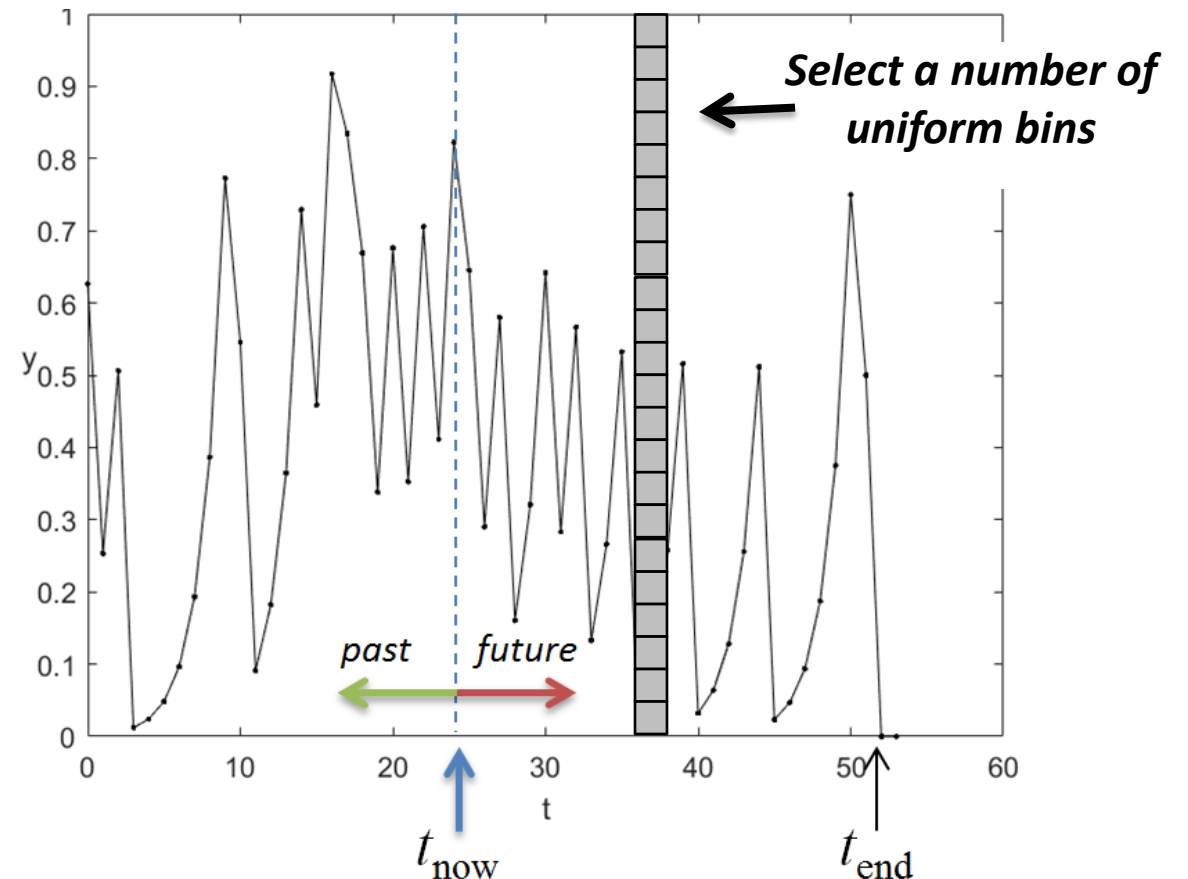
-  Discussion Paper # 1: Partially-Observed Markov Decision Processes  
-  Discussion Paper # 2: Prediction, Complexity and Learning  
-  Discussion Paper # 3: Gaussian Process for Time Series  
-  Discussion Paper # 4: Data-driven discovery of partial differential equations  
-  Paper of Interest: Information Bottleneck  
-  Background: Summary of Lie Groups  
-  Time Series Ensemble
-  1-d Time Series Exercise  

Exercise: 1- d Non-Stationary Time Series

Objective:

- Density estimation on low- D
- Learn the *states*
- Learning the dynamics
 - Gaussian Process
 - Information Bottleneck
- Calculate the Predictive Information
- Embed dynamics on a manifold
- Density estimation in high- D

An ensemble of 10^5 time series that all terminate with zero.



Pulling Back **DATA Resolution** into the MODEL Space

Size of Bins!

Sensors should be capable of observing much more than they will.

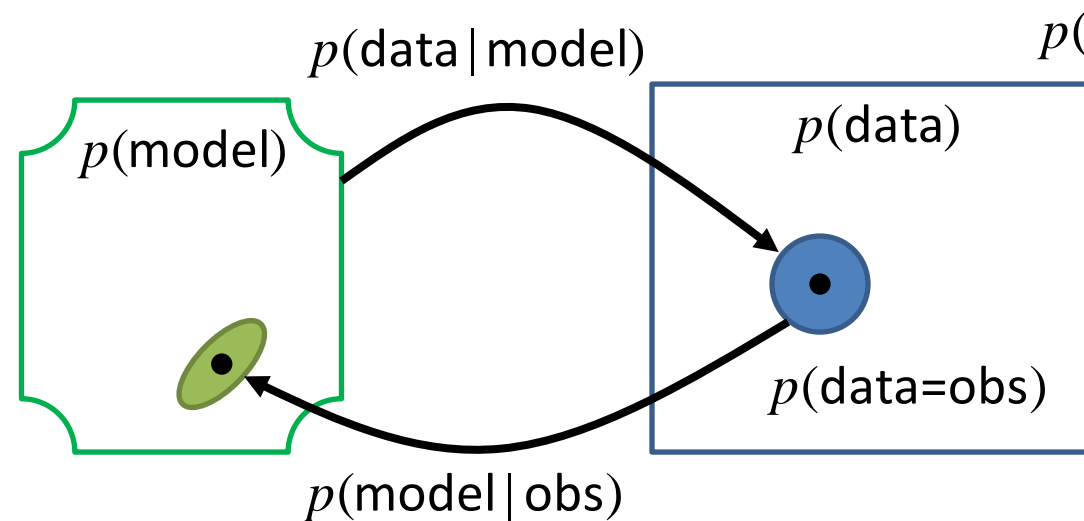
Always recognize this difference between what *could* have been observed and what was *actually* observed.

It is in this difference that models emerge as constraints on what we do observe in the physical world ...

“Ball in a Box” Analogy

- What could be observed? → How big is the box?
- What is observed? → Where is the ball?
- What is the resolution of the observation? → How big is the ball?

Bayesians believe they can more easily specify the size of the “Model” box using a prior distribution and this specifies the sampling distribution.

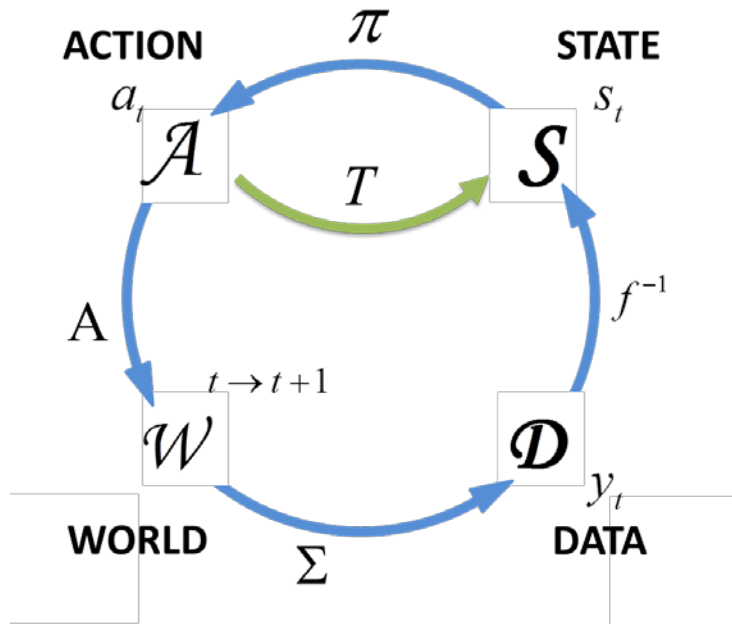


Frequentists believe they can more easily specify the size of the “Data” box using a sampling distribution and then test the significance of models.

Bayesians “pull back” the observations to the model space as a posterior probability.

NOTE: DATA space refers to the space of all *possible* observations.

Dynamical System / Control Theory / Agent Model



Model State:

$$p(s_t)$$

Dynamics Model:

$$T = p(s_{t+1} | a_t, s_t)$$

Observation Model:

$$f = p(y_t | s_t)$$

Policy:

$$a_t = \pi[p(s_t)]$$

$$p(s_t | y_t, a_{t-1}, s_{t-1}) = \frac{\overbrace{p(y_t | s_t)}^f \cdot \overbrace{p(s_t | a_{t-1}, s_{t-1})}^T}{\underbrace{p(y_t | a_{t-1}, s_{t-1})}}$$

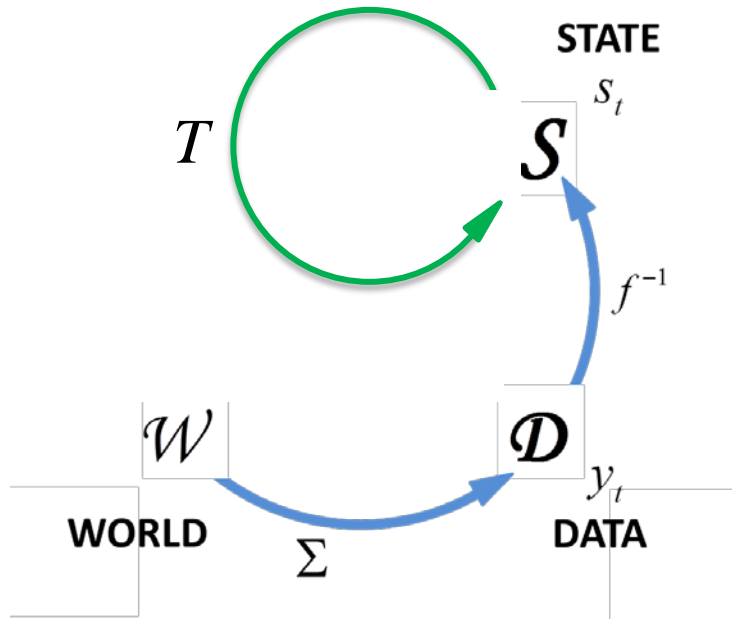


State estimation:

$$p(s_t | y_t, a_{t-1}) = \sum_{s_{t-1} \in \mathcal{S}} p(s_t | y_t, a_{t-1}, s_{t-1}) \cdot p(s_{t-1})$$

$$= \sum_{s_{t+1} \in \mathcal{S}} p(y_{t+1} | s_{t+1}) \cdot p(s_{t+1} | a_t, s_t)$$

Dynamical System / ~~Control Theory~~ / ~~Agent Model~~



Model State:

$$p(s_t)$$

Dynamics Model:

$$T = p(s_{t+1} | s_t)$$

Observation Model:

$$f = p(y_t | s_t)$$

~~**Policy:**~~

~~$$a_t = \pi[p(s_t)]$$~~

$$p(s_t | y_t, s_{t-1}) = \frac{\overbrace{p(y_t | s_t)}^f \cdot \overbrace{p(s_t | s_{t-1})}^T}{\underbrace{p(y_t | s_{t-1})}}$$



State estimation:

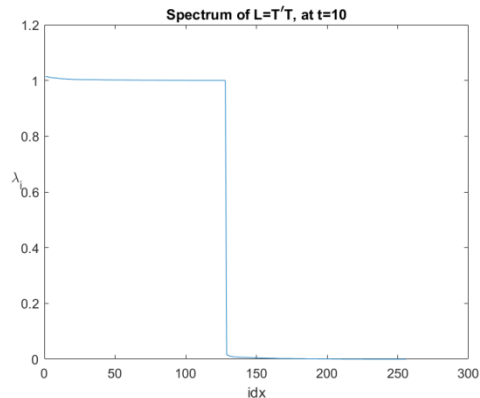
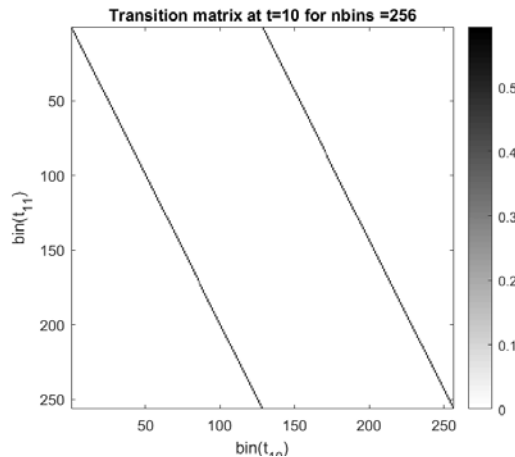
$$p(s_t | y_t) = \sum_{s_{t-1} \in \mathcal{S}} p(s_t | y_t, s_{t-1}) \cdot p(s_{t-1})$$

$$= \sum_{s_{t+1} \in \mathcal{S}} p(y_{t+1} | s_{t+1}) \cdot p(s_{t+1} | s_t)$$

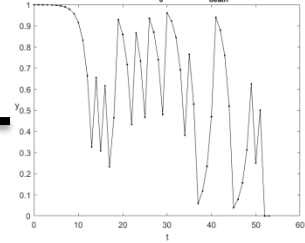
So, for ~~Gaussian processes~~ we have a recipe

Step 3

$$T = p(s_{t+1} | s_t)$$



1) Estimate the correlation function. ←



2) Take the inverse to get the kernel which appears in the probability distribution.

3) Isolate the coupling between past and future.

4) Look at the SVD of this coupling matrix.

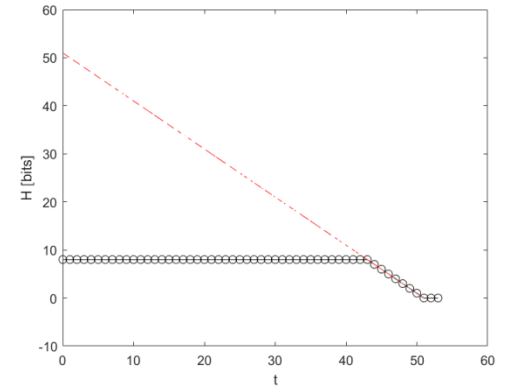
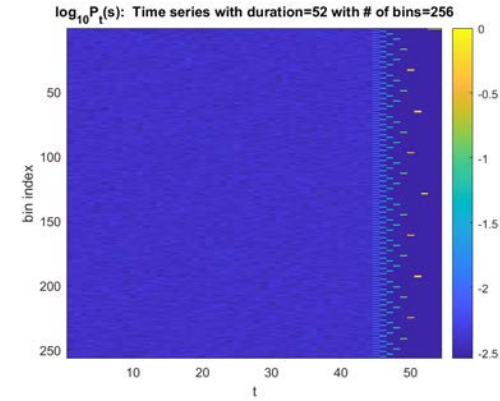
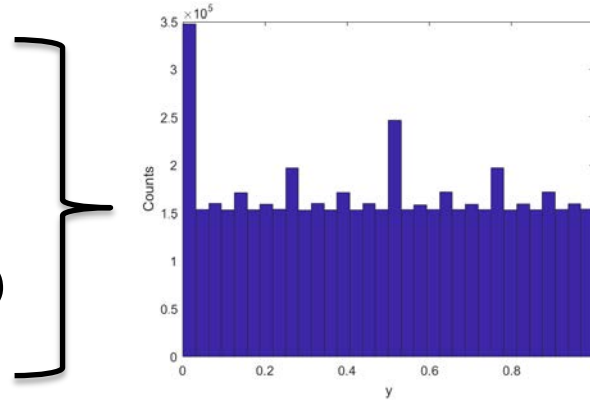
5) Dimensionality = # of nonzero singular values.

Importantly, if correlations decay slowly enough, this dimensionality can grow with the size of our analysis window.

Outline of steps in time-series exercise

Step 1: $p(y) \simeq p(s)$

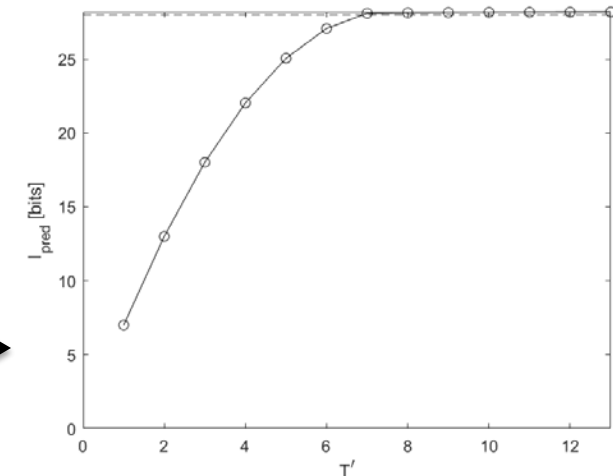
Step 2: $p(y_t) \simeq p(s_t)$



Step 3: $p(s_{t+1} | s_t) \rightarrow p(s_{t+i} | s_t) = \sum_{s_{t+1}, \dots, s_{t+i-1}} p(s_{t+i} | s_{t+i-1}) \times \dots \times p(s_{t+1} | s_t)$

Step 4: $I(S_t, S_{t+i}) = \sum_{s_t} \sum_{s_{t+i}} p(s_{t+i} | s_t) p(s_t) \log_2 \frac{p(s_{t+i} | s_t)}{p(s_{t+i})}$

$$I_{pred}(t, t+n) = \sum_{i=1}^n I(S_t, S_{t+i}) \quad \longrightarrow$$



Our Friday Discussion Paper

Data-driven discovery of partial differential equations

Samuel H. Rudy^{1*}, Steven L. Brunton², Joshua L. Proctor³, and J. Nathan Kutz¹

¹ *Department of Applied Mathematics, University of Washington, Seattle, WA. 98195*

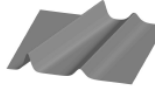
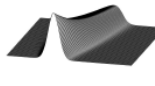

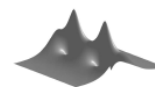
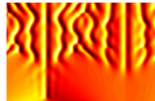
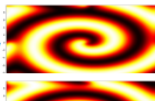
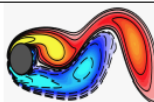
² *Department of Mechanical Engineering, University of Washington, Seattle, WA. 98195 and*

³ *Institute for Disease Modeling, , 3150 139th Ave SE, Bellevue, WA 98005*

(Dated: September 22, 2016)

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

PACS numbers: 05.45.-a, 05.45.Yv

PDE	Form
 KdV	$u_t + 6uu_x + u_{xxx} = 0$
 Burgers	$u_t + uu_x - \epsilon u_{xx} = 0$
 Schrödinger	$iu_t + \frac{1}{2}u_{xx} - \frac{x^2}{2}u = 0$
 NLS	$iu_t + \frac{1}{2}u_{xx} + u ^2u = 0$
 KS	$u_t + uu_x + u_{xx} + u_{xxxx} = 0$
 Reaction Diffusion	$u_t = 0.1\nabla^2 u + \lambda(A)u - \omega(A)v$ $v_t = 0.1\nabla^2 v + \omega(A)u + \lambda(A)v$ $A^2 = u^2 + v^2, \omega = -\beta A^2, \lambda = 1 - A^2$
 Navier Stokes	$\omega_t + (\mathbf{u} \cdot \nabla)\omega = \frac{1}{Re}\nabla^2\omega$

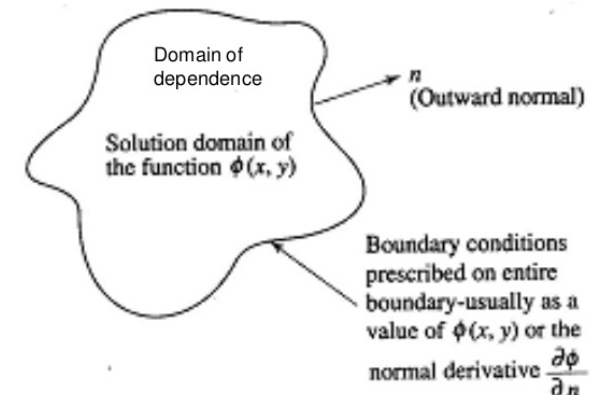
Models: Partial Differential Equations

- The *Ultimate* Local Model: PDE's are models constructed as constraint equations among local operators (partial derivatives).
- Finding the PDE: These constraints are often related to *local* conservation principles, symmetries, or requirements of non-negativity. Noether's Theorem, continuous symmetries → conservation laws.
- Going non-local: A PDE-based model can be converted into a non-local model via the integration of a kernel. But a key question concerns whether an apparently non-local model can be reformulated as a PDE-based model.
- Specifying the Model Domain: In either case, the challenge is incorporating the effect of external forcing functions, boundary and initial conditions or the asymptotic steady-state of the system.

$$f(x, t) \quad \frac{\partial}{\partial t} \quad \frac{\partial}{\partial x^i} \quad \frac{\partial}{\partial x^i} x^j \frac{\partial}{\partial x^j} \quad \frac{\partial^2}{\partial x^i \partial x^j}$$

$$\frac{\partial u}{\partial t} = \mathcal{N} \left(f, \frac{\partial}{\partial x^i}, \frac{\partial^2}{\partial x^i \partial x^j}, \dots \right)$$

$$G^{RL}(\tau, \tau') = \int_C d\tau_1 g^R(\tau - \tau_1) V^{RL} g^L(\tau_1 - \tau') \\ + \int_C d\tau_1 \int_C d\tau_2 g^R(\tau - \tau_1) V^{RL} g^L(\tau_1 - \tau_2) V^{LR} G^{RL}(\tau_2, \tau')$$



Learning a PDE as Regression

In what follows, we consider a PDE of the form

$$\mathbf{y} = f(\mathbf{x}) \longrightarrow u_t = N(u, u_x, u_{xx}, \dots, x, \mu) \quad (1)$$

where the subscripts denote partial differentiation in either time or space, and $N(\cdot)$ is an unknown right-hand

The PDE in this library is:

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

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

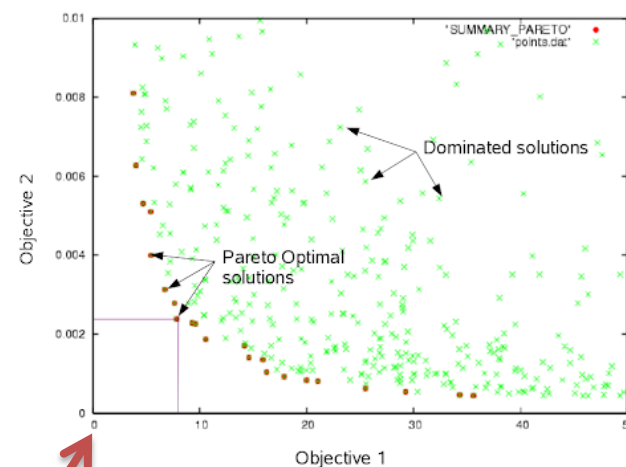
Learning a PDE as Regression

Instead, we approximate the problem using candidate solutions to a ridge regression problem with hard thresholding, which we call sequential threshold ridge regression (STRidge in Algorithm 1). For a given tolerance and λ , this gives a sparse approximation to ξ . We iteratively refine the tolerance of Algorithm 1 to find the best predictor based on the selection criteria,

$$\hat{\xi} = \operatorname{argmin}_{\xi} \|\Theta(\mathbf{U}, \mathbf{Q})\xi - \mathbf{U}_t\|_2^2 + \underbrace{\epsilon \kappa(\Theta(\mathbf{U}, \mathbf{Q})) \|\xi\|_0}_{\text{Regularization to control complexity } \sim \log(\text{prior})} \quad (3)$$

where $\kappa(\Theta)$ is the condition number of the matrix Θ , indicating stronger regularization for ill-posed problems. Penalizing $\|\xi\|_0$ discourages over fitting by selecting from the optimal position in a Pareto front.

Regularization to control complexity
 $\sim \log(\text{prior})$



PDE-FIND Algorithm

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

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



PDE Function Identification of Nonlinear Dynamics

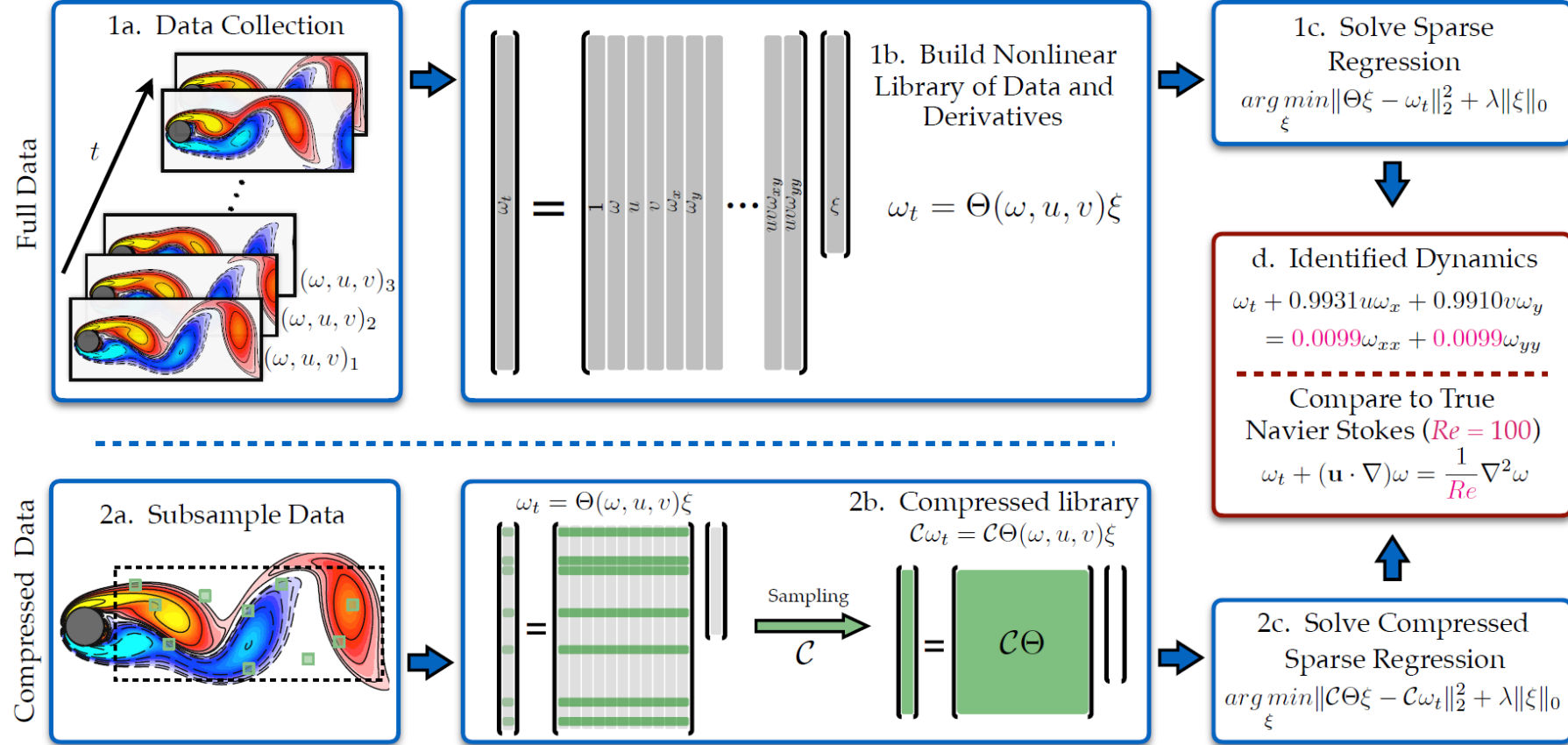


FIG. 1: Steps in the PDE functional identification of nonlinear dynamics (PDE-FIND) algorithm, applied to infer the Navier-Stokes equation from data. **1a.** Data is collected as snapshots of a solution to a PDE. **1b.** Numerical derivatives are taken and data is compiled into a large matrix Θ , incorporating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

Results of PDE-FIND algorithm

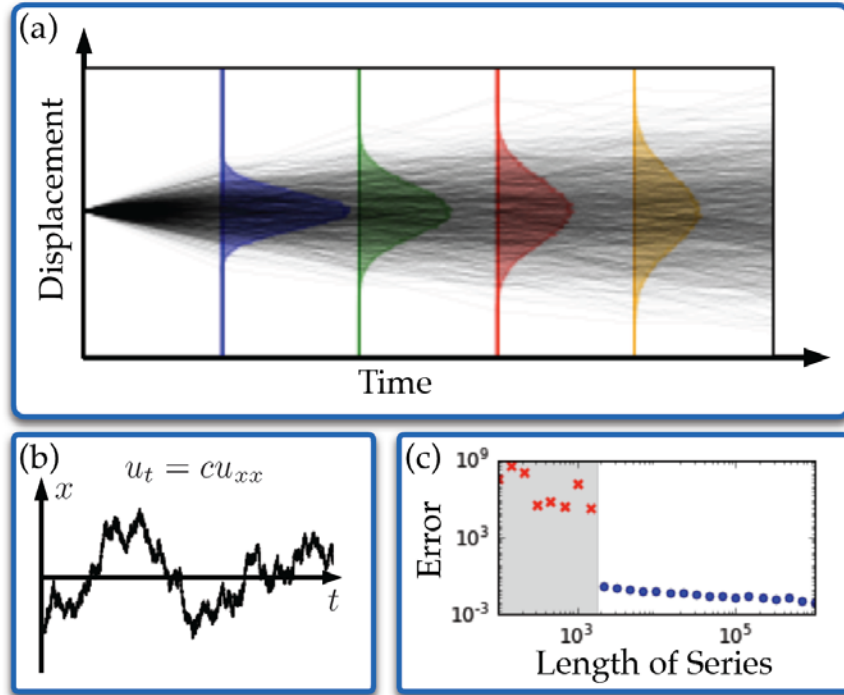


FIG. 2: Inferring the diffusion equation from a single Brownian motion. (a) Time series is broken into many short random walks that are used to construct histograms of the displacement. (b) The Brownian motion trajectory, following the diffusion equation. (c) Parameter error ($\|\xi^* - \hat{\xi}\|_1$)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

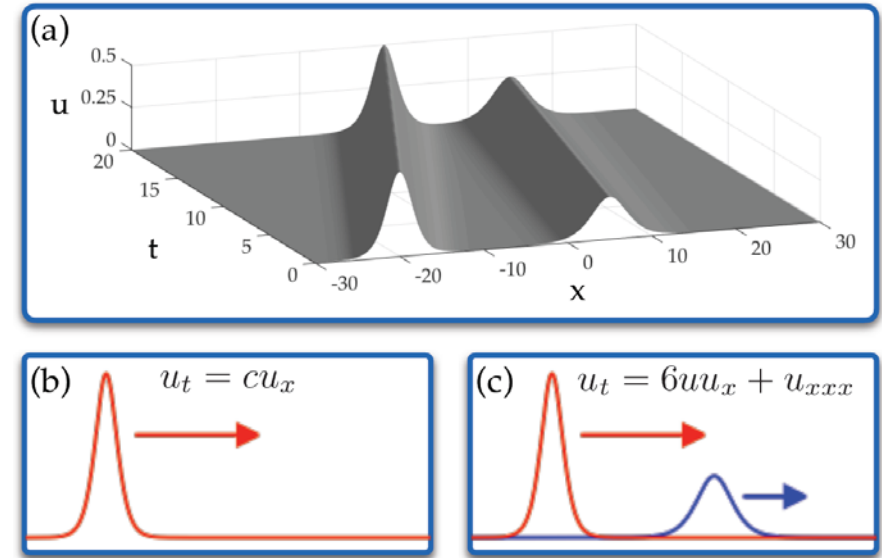


FIG. 3: Inferring nonlinearity via observing solutions at multiple amplitudes. (a) An example 2-soliton solution to the KdV equation. (b) Applying our method to a single soliton solution determines that it solves the standard advection equation. (c) Looking at two completely separate solutions reveals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

Results of PDE-FIND algorithm

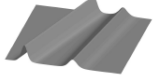
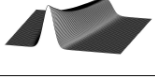

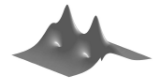
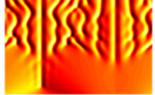
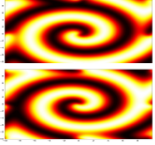
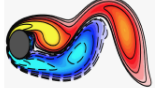
PDE	Form	Error (no noise, noise)	Discretization
 KdV	$u_t + 6uu_x + u_{xxx} = 0$	$1\% \pm 0.2\%, 7\% \pm 5\%$	$x \in [-30, 30], n=512, t \in [0, 20], m=201$
 Burgers	$u_t + uu_x - \epsilon u_{xx} = 0$	$0.15\% \pm 0.06\%, 0.8\% \pm 0.6\%$	$x \in [-8, 8], n=256, t \in [0, 10], m=101$
 Schrödinger	$iu_t + \frac{1}{2}u_{xx} - \frac{x^2}{2}u = 0$	$0.25\% \pm 0.01\%, 10\% \pm 7\%$	$x \in [-7.5, 7.5], n=512, t \in [0, 10], m=401$
 NLS	$iu_t + \frac{1}{2}u_{xx} + u ^2u = 0$	$0.05\% \pm 0.01\%, 3\% \pm 1\%$	$x \in [-5, 5], n=512, t \in [0, \pi], m=501$
 KS	$u_t + uu_x + u_{xx} + u_{xxxx} = 0$	$1.3\% \pm 1.3\%, 70\% \pm 27\%$	$x \in [0, 100], n=1024, t \in [0, 100], m=251$
 Reaction Diffusion	$u_t = 0.1\nabla^2 u + \lambda(A)u - \omega(A)v$ $v_t = 0.1\nabla^2 v + \omega(A)u + \lambda(A)v$ $A^2 = u^2 + v^2, \omega = -\beta A^2, \lambda = 1 - A^2$	$0.02\% \pm 0.01\%, 3.8\% \pm 2.4\%$	$x, y \in [-10, 10], n=256, t \in [0, 10], m=201$ subsample 1.14%
 Navier Stokes	$\omega_t + (\mathbf{u} \cdot \nabla)\omega = \frac{1}{Re}\nabla^2\omega$	$1\% \pm 0.2\%, 7\% \pm 6\%$	$x \in [0, 9], n_x=449, y \in [0, 4], n_y=199,$ $t \in [0, 30], m=151, \text{subsample } 2.22\%$

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