Data-driven discovery for geophysical systems: Integrating machine learning and dynamical systems for learning multi-scale physical systems

- Physical Model discovery
- Coordinates, manifolds & Embeddings
- Measurement & Sensors

J. Nathan Kutz
Department of Applied Mathematics
University of Washington
Email: **kutz@uw.edu**

Steven Brunton
Mechanical Engineering
University of Washington

SIAM Geosciences 2019 - Mini Tutorial

Question #1

What is the nature of your data?

- quality
- quality
- observability
- extrapolation vs interpolation

Mathematical Framework

Dynamics

State-space

Parameters

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}, t, \Theta, \Omega)$$

Dynamics

Stochastic effects

Measurement

$$\mathbf{y}(t_k) = h(t_k, \mathbf{x}(t_k), \Xi)$$

Measurement model

Measurement noise

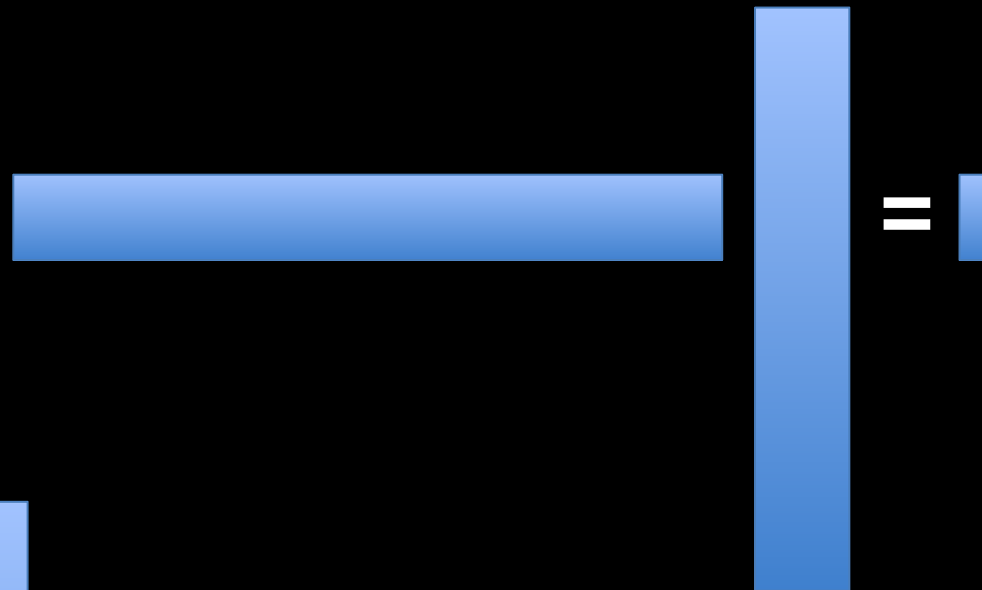
Model Discovery

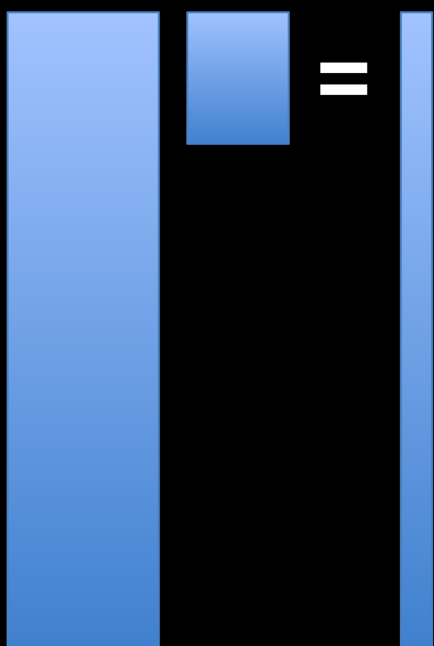
Finding governing equations

$$Ax=b$$

Data Science Today

Under





Over

- \
- pinv
- Lasso
- Ridge
- Elastic net
- Robust fit

$$Ax=b$$

subject to

$$\min g(x)$$

$$f(A,x)=b$$

subject to

$$\min g(x)$$

Governing Dynamical Systems

Generic nonlinear , time-dependent, parametric system

$$\frac{d\mathbf{x}}{dt} = N(\mathbf{x}, t; \mu)$$

Measurements (assimilation)

$$G(\mathbf{x}, t_k) = 0$$

What Could the Right Side Be?

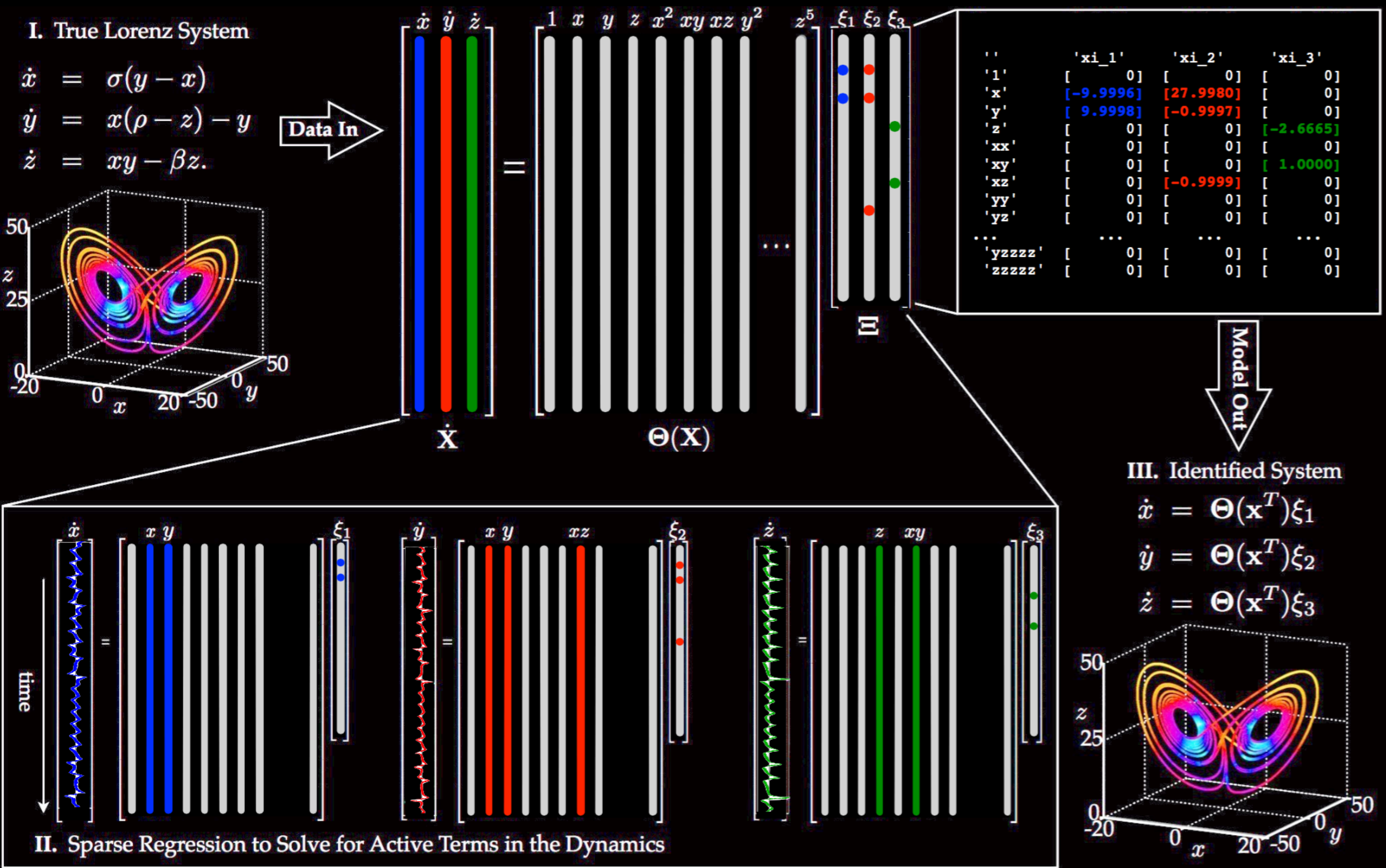
Limited by your imagination

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

2nd degree polynomials

$$\mathbf{X}^{P_2} = \begin{bmatrix} x_1^2(t_1) & x_1(t_1)x_2(t_1) & \cdots & x_2^2(t_1) & x_2(t_1)x_3(t_1) & \cdots & x_n^2(t_1) \\ x_1^2(t_2) & x_1(t_2)x_2(t_2) & \cdots & x_2^2(t_2) & x_2(t_2)x_3(t_2) & \cdots & x_n^2(t_2) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_1^2(t_m) & x_1(t_m)x_2(t_m) & \cdots & x_2^2(t_m) & x_2(t_m)x_3(t_m) & \cdots & x_n^2(t_m) \end{bmatrix}$$

Sparse Identification of Nonlinear Dynamics (SINDy)



Nonlinear Systems ID

1. Collect Data

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

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

2. Build Library of Candidate Nonlinearities

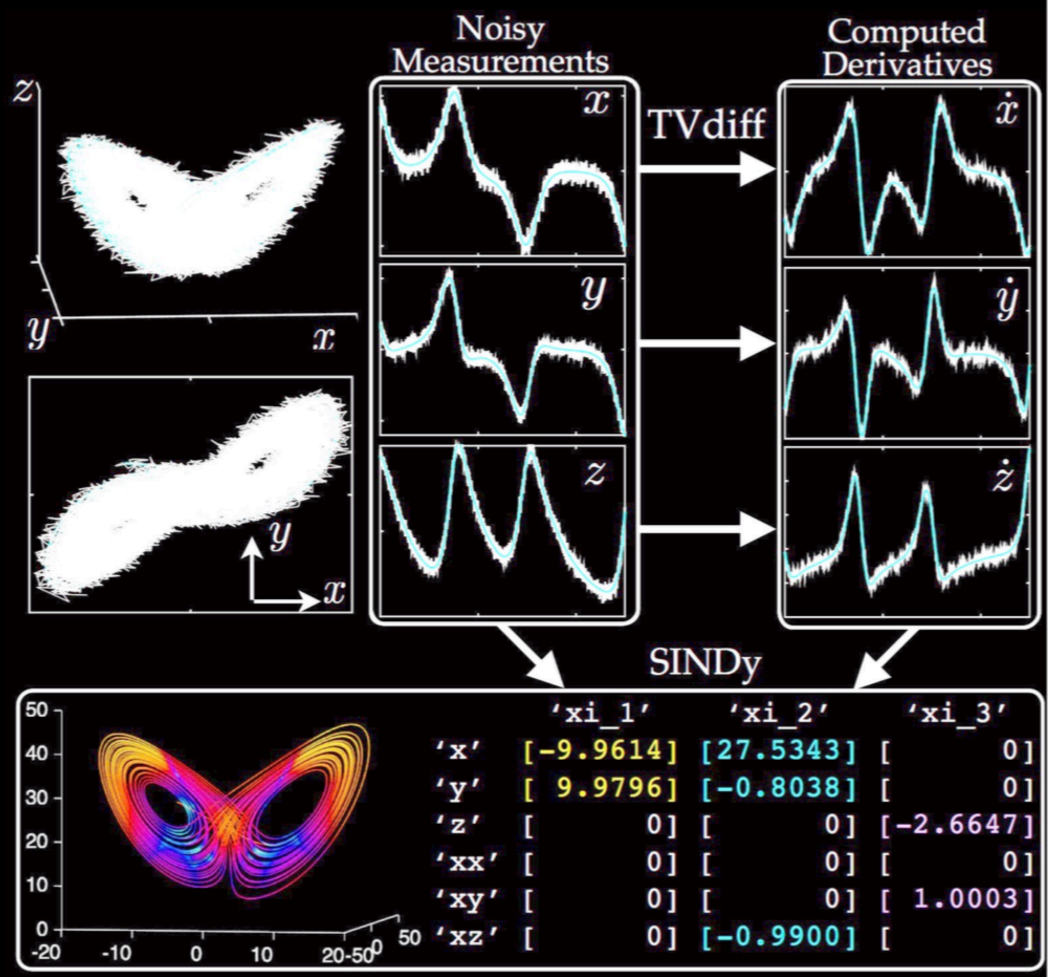
$$\Theta(\mathbf{X}) = \begin{bmatrix} | & | & | & | & & | & | & \\ 1 & \mathbf{X} & \mathbf{X}^{P_2} & \mathbf{X}^{P_3} & \cdots & \sin(\mathbf{X}) & \cos(\mathbf{X}) & \cdots \\ | & | & | & | & & | & | & \end{bmatrix}.$$

3. Sparse Regression to Find Active Terms

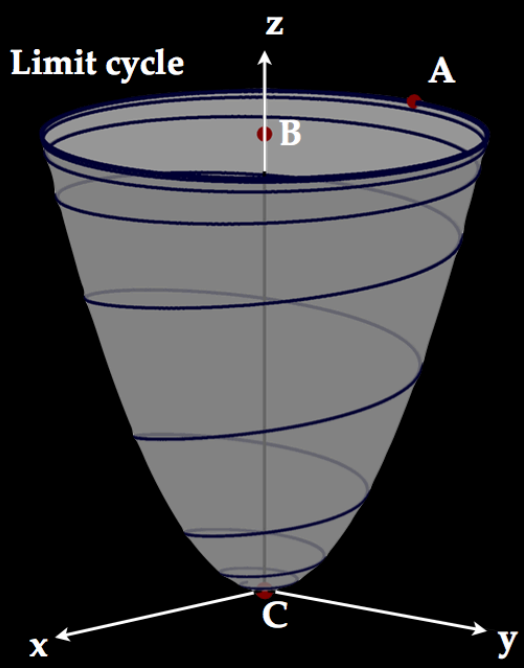
$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi.$$

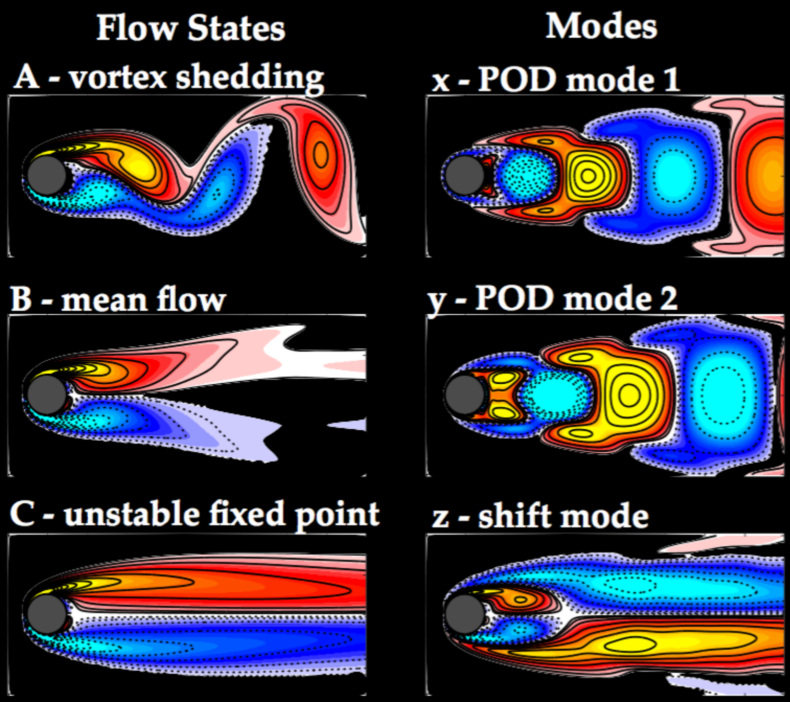
4. Nonlinear Model

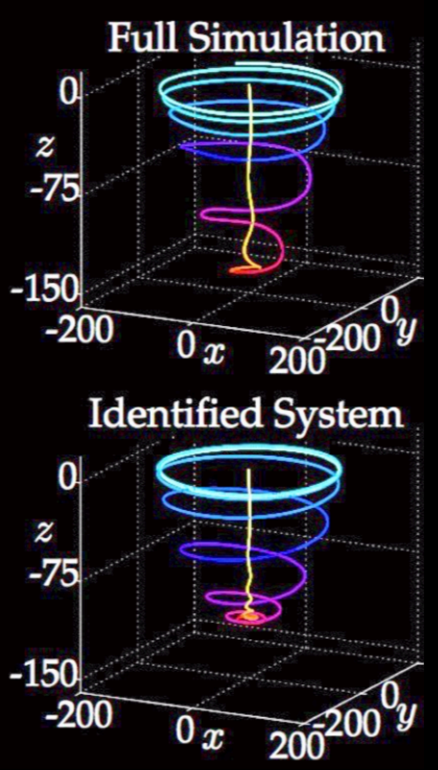
$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) = \Xi^T(\Theta(\mathbf{x}^T))^T$$



Identifying Slow Manifolds







30 years of progress

$$
\begin{aligned}
\dot{x} &= \mu x - \omega y + Axz \\
\dot{y} &= \omega x + \mu y + Ayz \\
\dot{z} &= -\lambda(z - x^2 - y^2).
\end{aligned}
$$

1. Hopf bifurcations as path to turbulence
Ruelle & Takens, *Communications in Mathematical Physics*, 1971

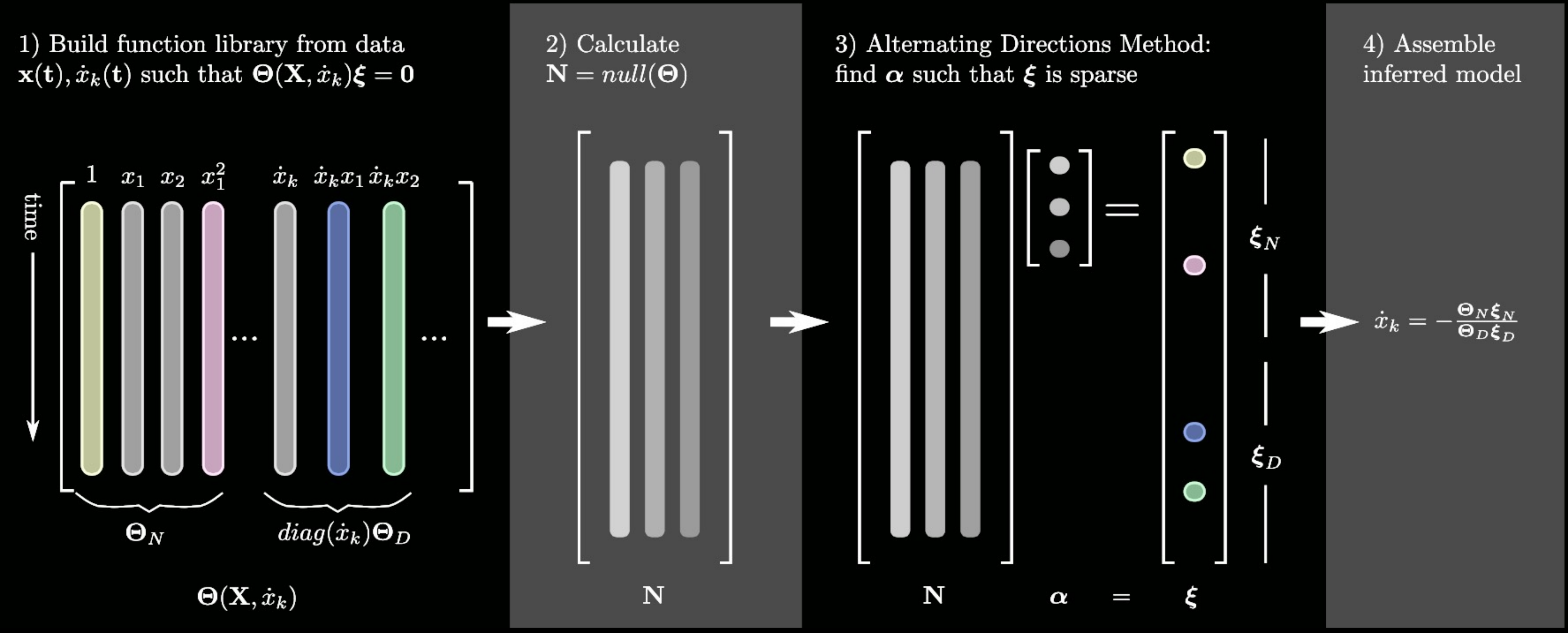
2. Vortex shedding and Hopf bifurcation
Jackson, *Journal of Fluid Mechanics, 1987.*

3. Mean-field model with slow manifold
Noack, Afanasiev, Morzynski, Tadmor, & Thiele,
Journal of Fluid Mechanics, 2003.

Modifications: Implicit-SINDy

Niall Mangan

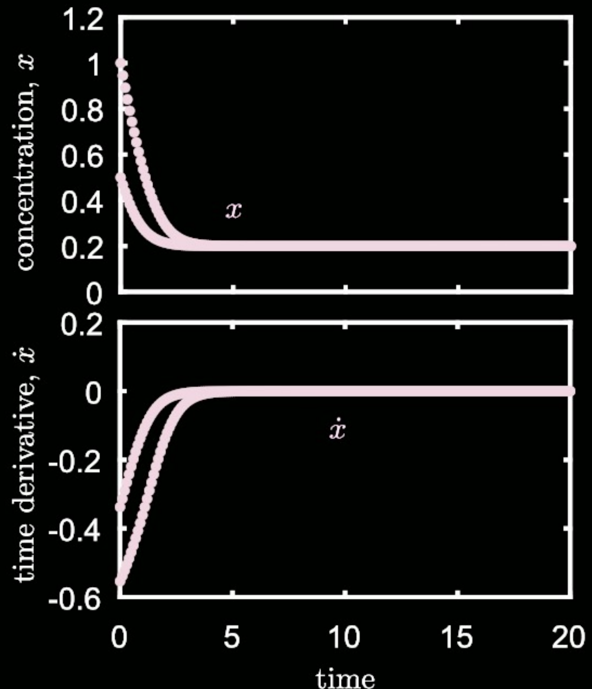
$$\dot{x}_k = \frac{f_N(\mathbf{x})}{f_D(\mathbf{x})}$$



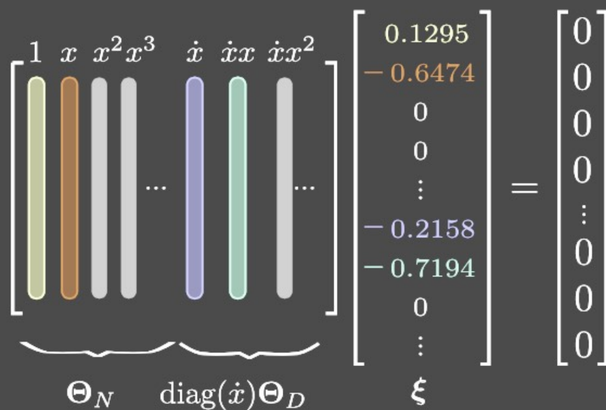
Michaelis-Menten: enzymatic reaction

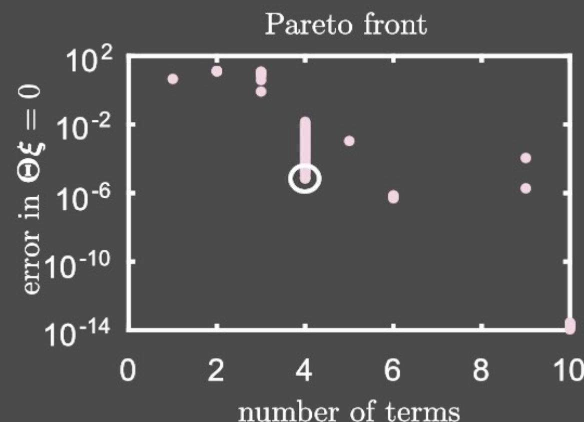
1) Generate test data from system:

$$\dot{x} = 0.6 - \frac{1.5x}{0.3 + x}$$



2) Build functional library. Sparsely select terms and find λ where error drops on Pareto front:



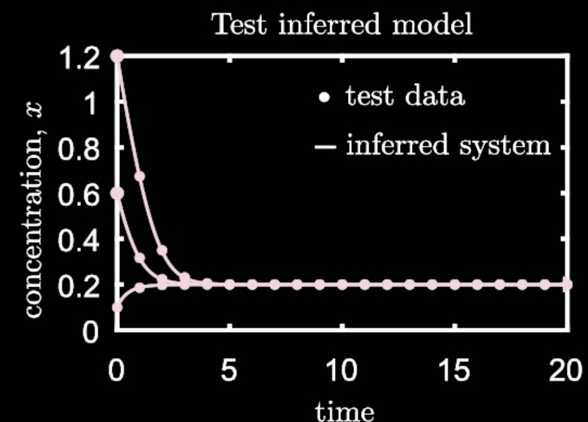


3) Construct inferred model and compare with data from new initial conditions:

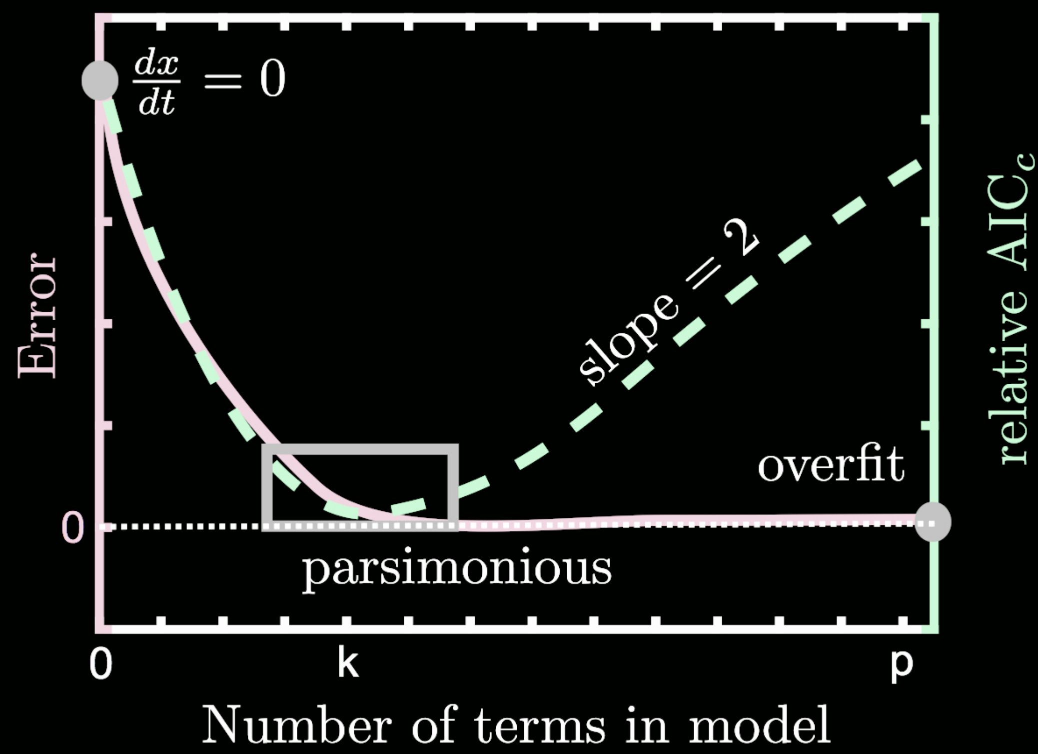
$$\dot{x} = \frac{0.1295 - 0.6474x}{0.2158 + 0.7194x} = \frac{0.6 - 3x}{1 + 3.33x}$$

$$= \frac{0.6(1 + 3.33x)}{1 + 3.33x} - \frac{1.999x + 3x}{1 + 3.33x}$$

$$= 0.6 - \frac{1.5x}{0.3 + x}$$



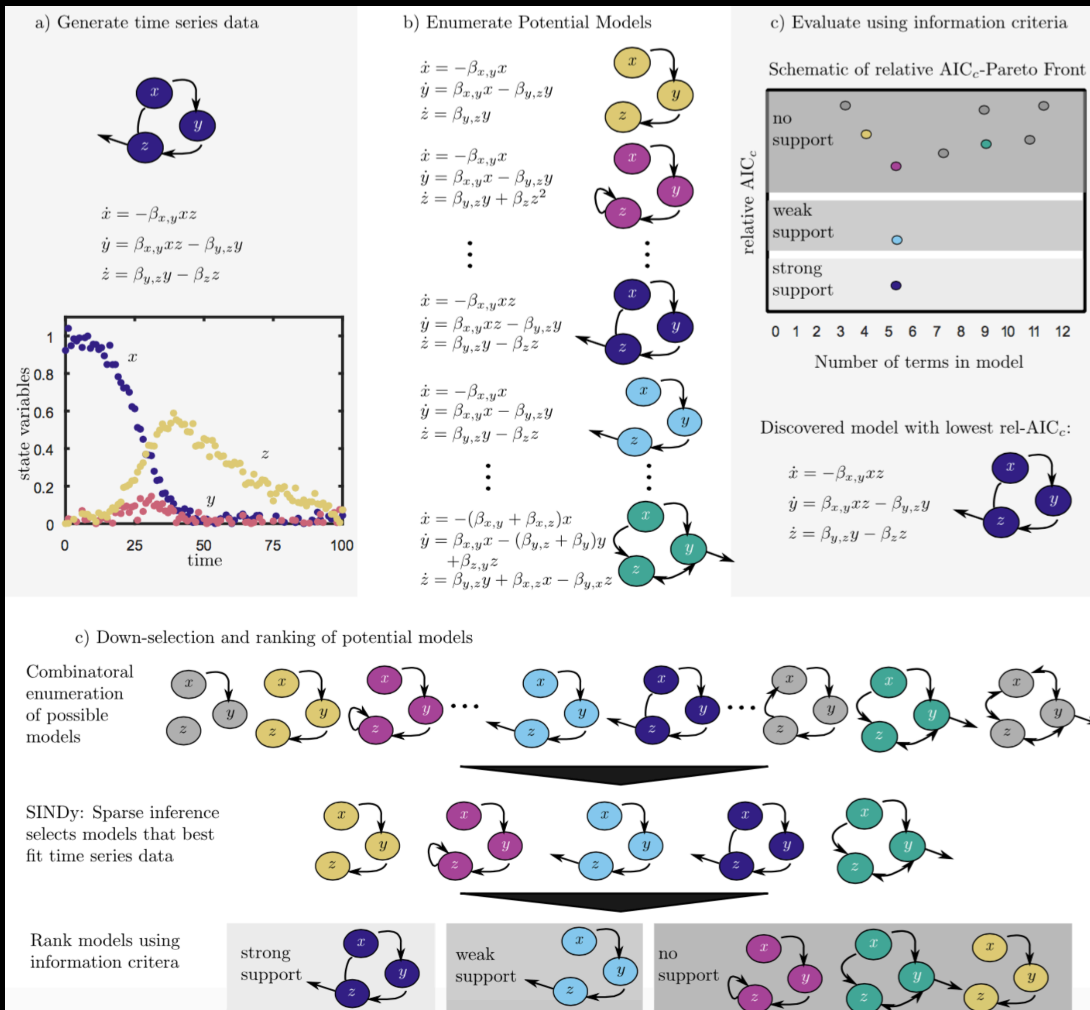
Parsimony



$$AIC_j = 2k - 2\ln(L(\mathbf{x}, \hat{\mu}))$$

$L(\mathbf{x}, \mu) = P(\mathbf{x}|\mu)$ is the likelihood function

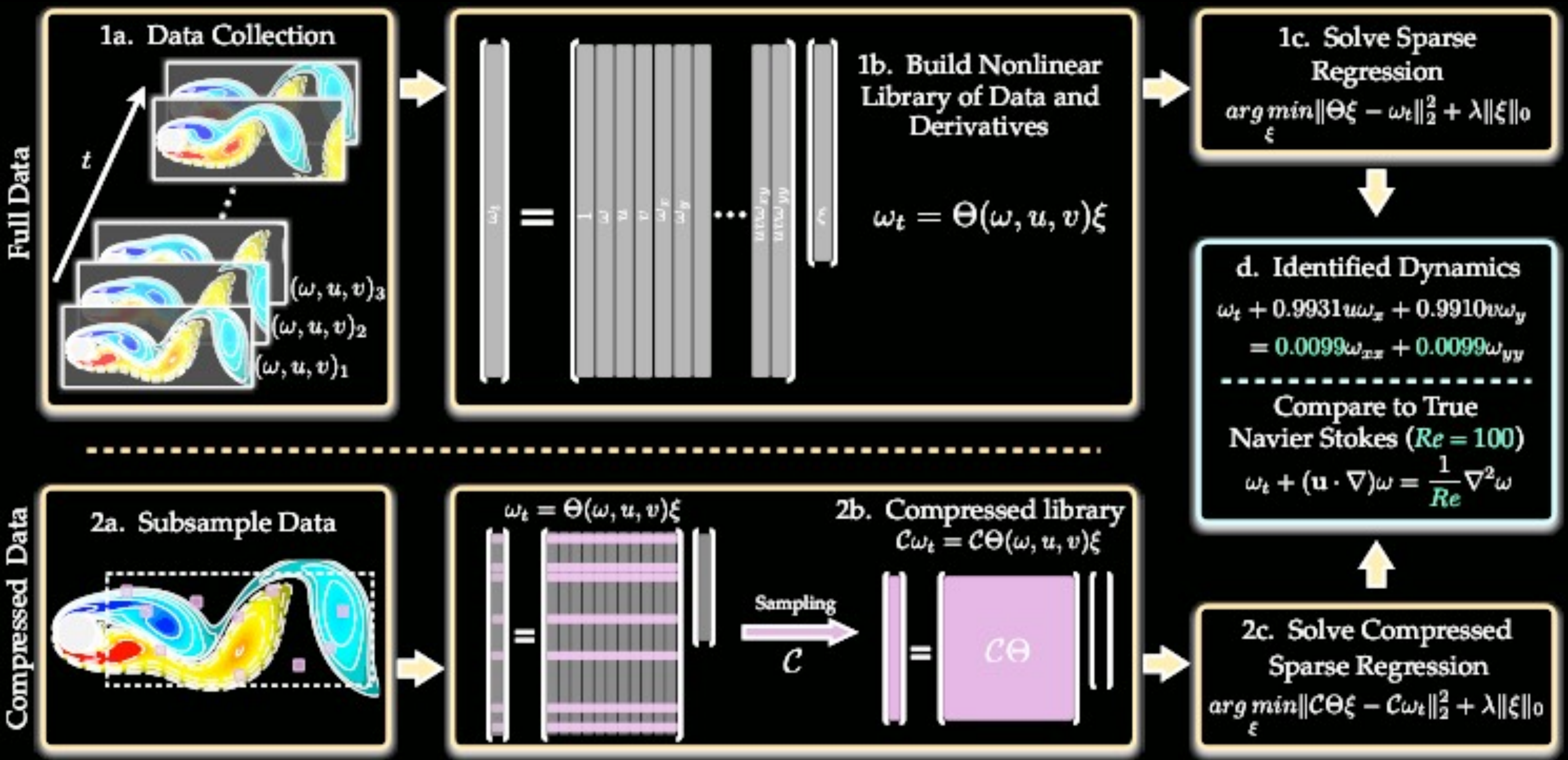
Model Selection and Information Theory



Model Selection

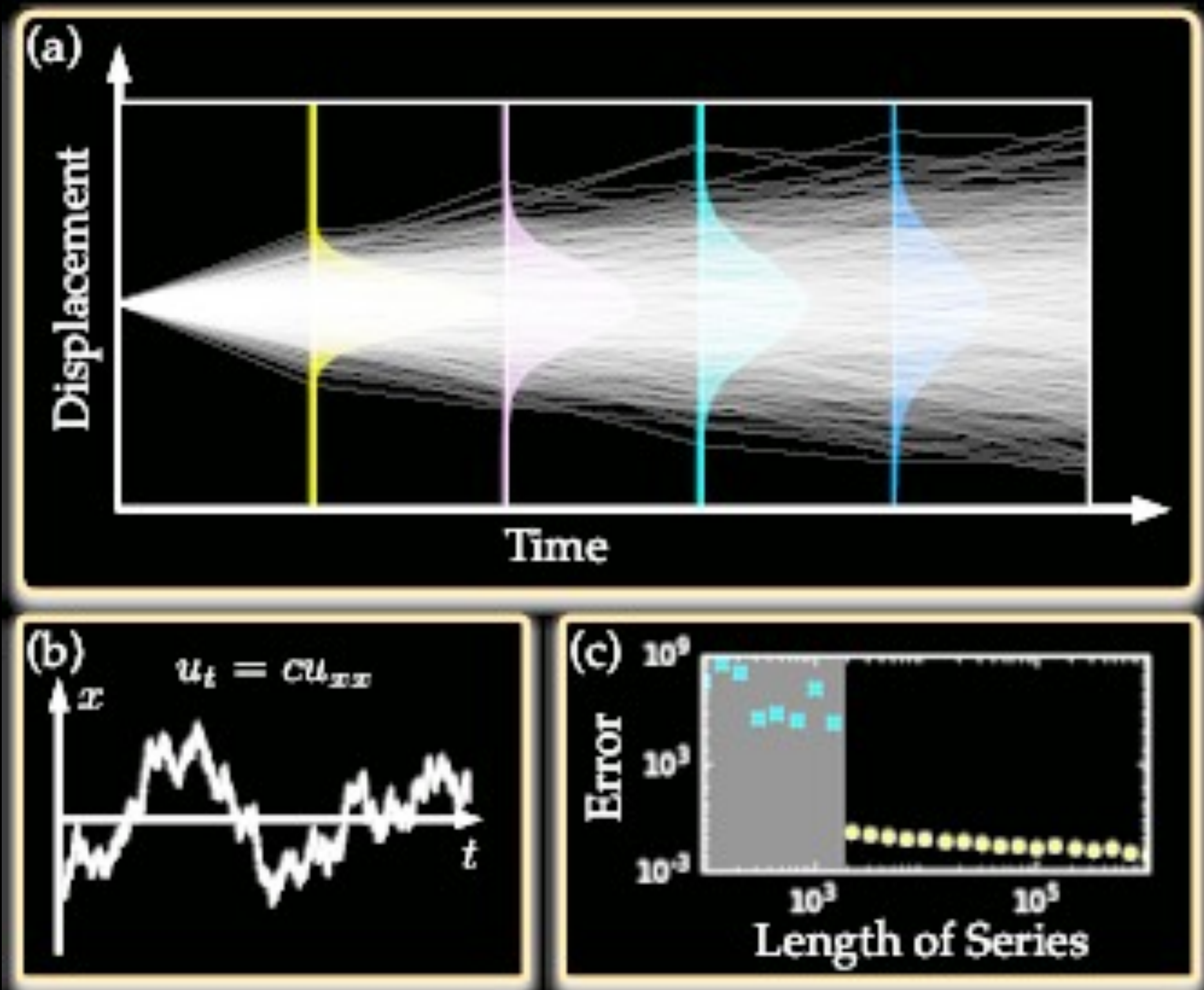
- **1950s KL divergence**
- **Early 70s AIC (Akaike)**
- **78 BIC (G. Schwarz)**
- **BIC/AIC limited # of models**

Discovering PDEs

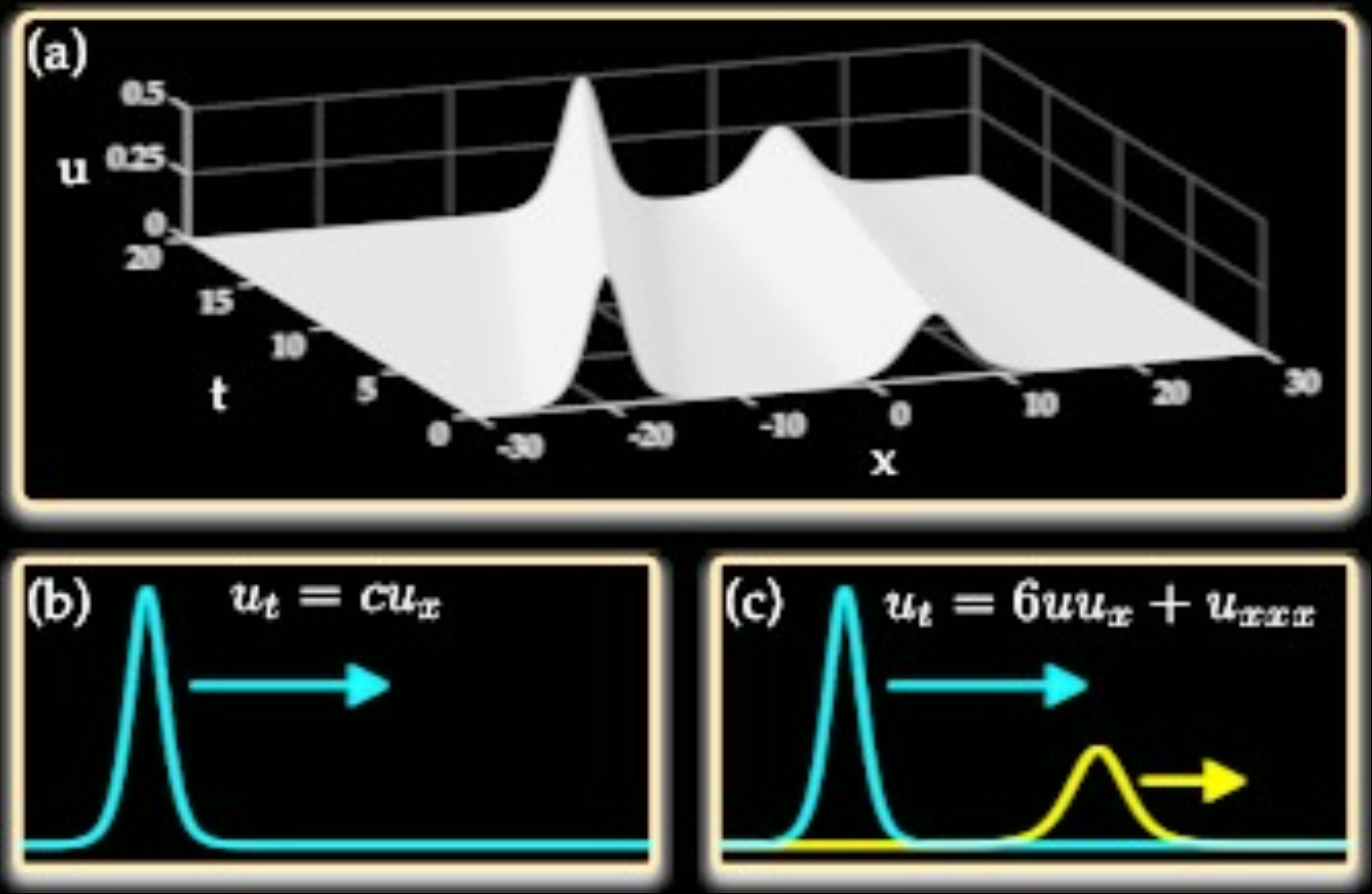


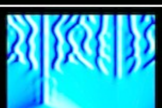
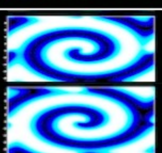
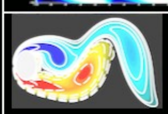
Sam Rudy

Lagrangian Measurements

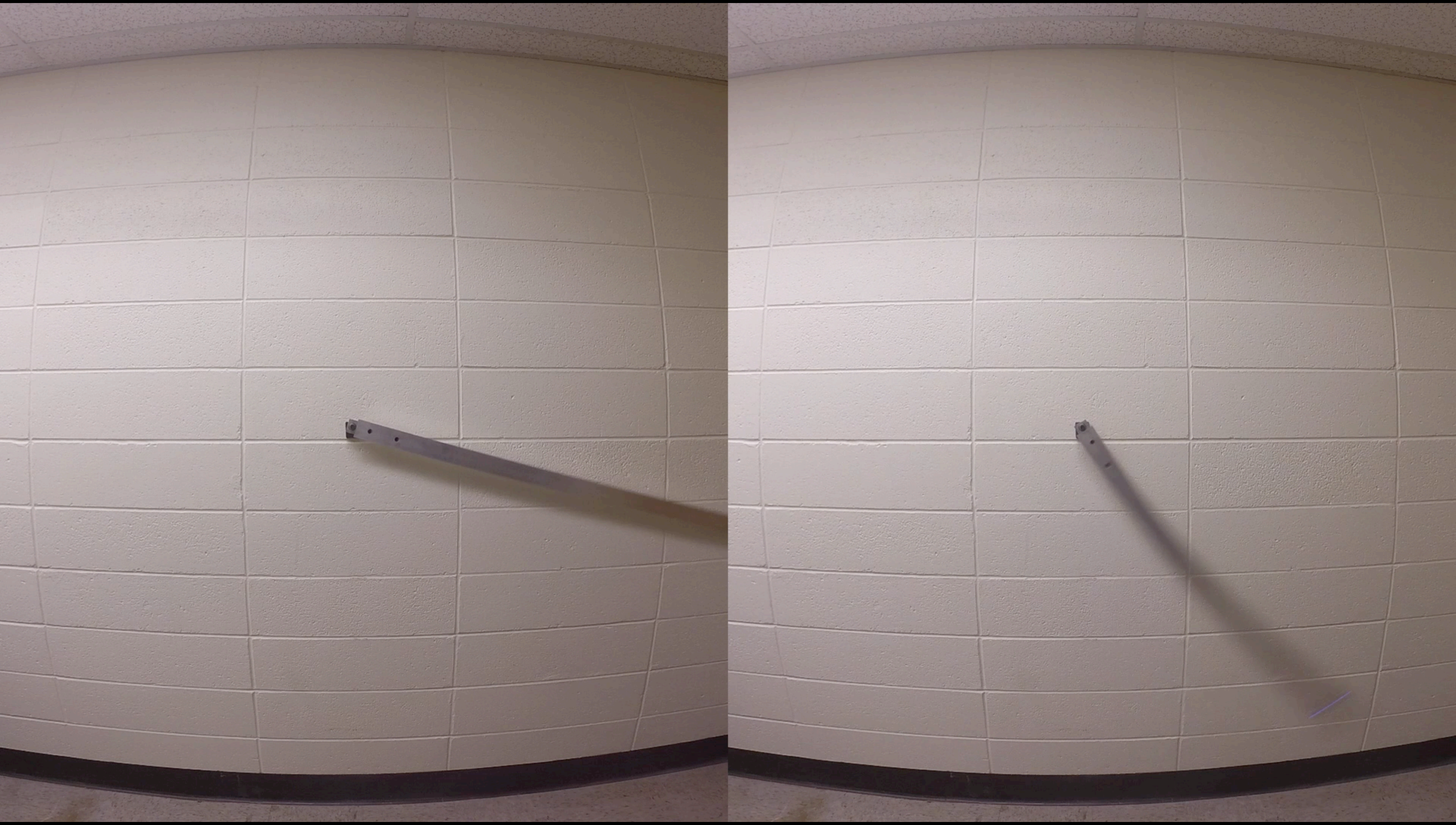


Disambiguation



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	$u_t + uu_x - \epsilon u_{xx} = 0$	$0.15\% \pm 0.06\%, 0.8\% \pm 0.6\%$	$x \in [-8, 8], n{=}256,\ t \in [0, 10], m{=}101$
 Schrodinger	$iu_t + \frac{1}{2}u_{xx} - \frac{x^2}{2}u = 0$	$0.25\% \pm 0.01\%, 10\% \pm 7\%$	$x \in [-7.5, 7.5], n{=}512,\ t \in [0, 10], m{=}401$
 NLS	$iu_t + \frac{1}{2}u_{xx} + \|u\|^2 u = 0$	$0.05\% \pm 0.01\%, 3\% \pm 1\%$	$x \in [-5, 5], n{=}512,\ t \in [0, \pi], m{=}501$
 KS	$u_t + uu_x + u_{xx} + u_{xxxx} = 0$	$1.3\% \pm 1.3\%, 70\% \pm 27\%$	$x \in [0, 100], n{=}1024,\ t \in [0, 100], m{=}251$
 R-D	$u_t = 0.1\nabla^2 u + \lambda(A)u - \omega(A)v$ $v_t = 0.1\nabla^2 v + \omega(A)u + \lambda(A)v$ $A{=}u^2{+}v^2, \omega{=}-\beta A^2, \lambda{=}1-A^2$	$0.02\% \pm 0.01\%,\ 3.8\% \pm 2.4\%$	$x, y \in [-10, 10], n{=}256,\ t \in [0, 10], m{=}201$ subsample $3 \cdot 10^5$
 Navier Stokes	$\omega_t + (\mathbf{u} \cdot \nabla)\omega = \frac{1}{Re}\nabla^2 \omega$	$1\% \pm 0.2\%$, $7\% \pm 6\%$	$x \in [0, 9], n_x{=}449,\ y \in [0, 4], n_y{=}199,$ $t \in [0, 30], m{=}151,$ subsample $3 \cdot 10^5$

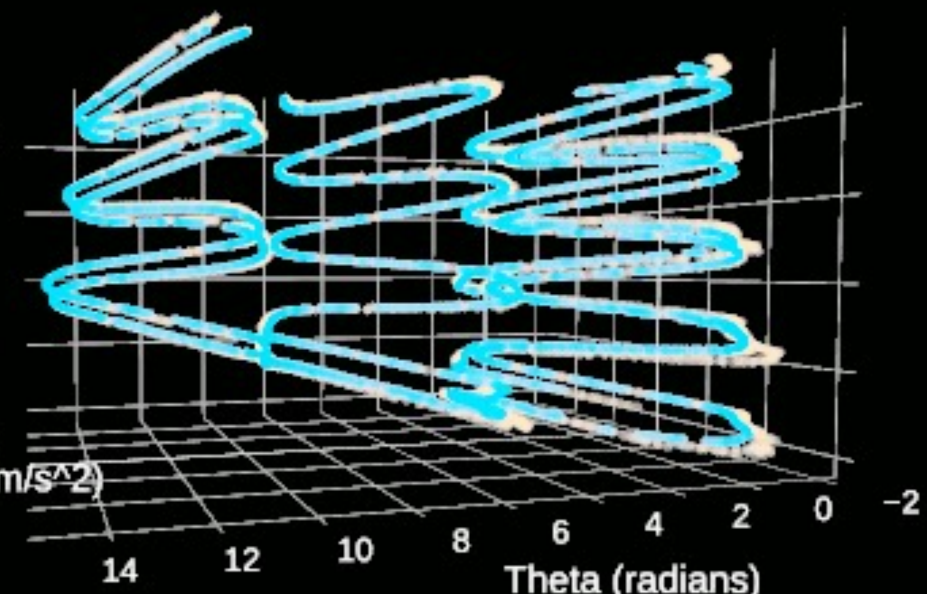
Experiments

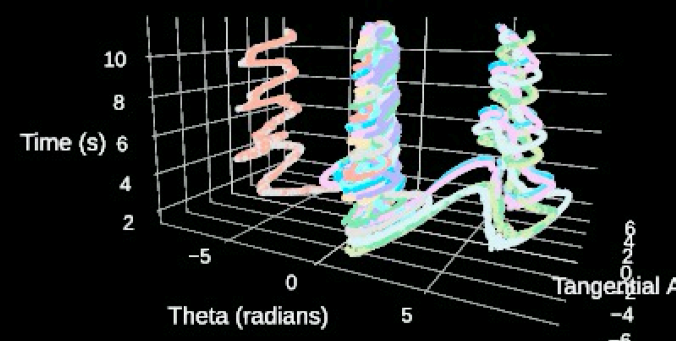


Arduino Magic

Taren Gorman

Data vs. SINDy Plot





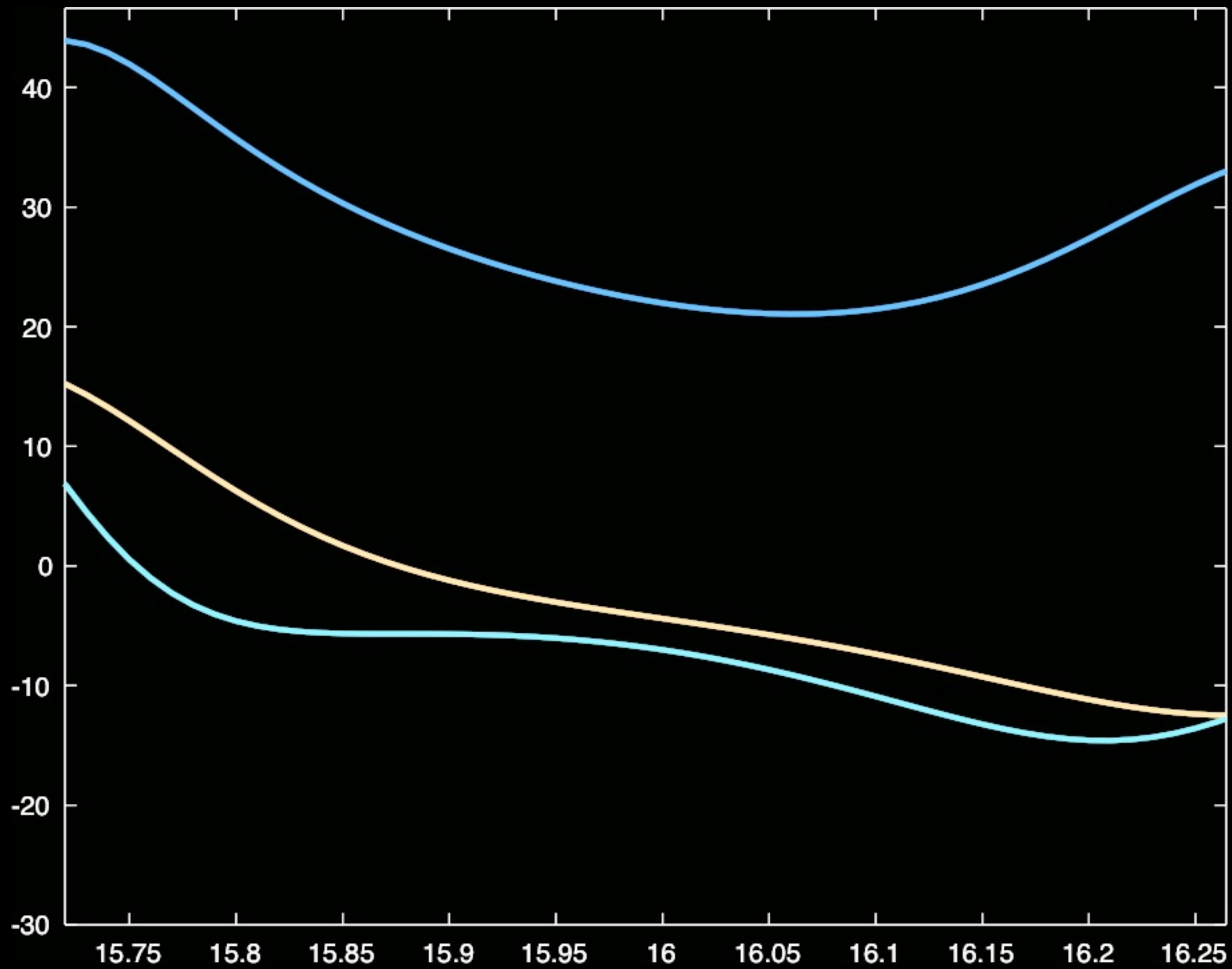

```
/home/taren/ana
ning:

divide by zero

/home/taren/ana
ning:

divide by zero

(77854, 2) (778
With  -1 jobs, fit and predict STRidge took 5.747981 seconds.
dx_0 / dt =  1.0*x_1
dx_1 / dt =  -0.1460697460858498*x_1+-3.9120253716489075*sin(x_0)
```

KEY CHALLENGES

- Limited measurements & data
- Noise
- Multi-scale physics
- Latent variables
- Parametric dependencies
- Stochastic systems

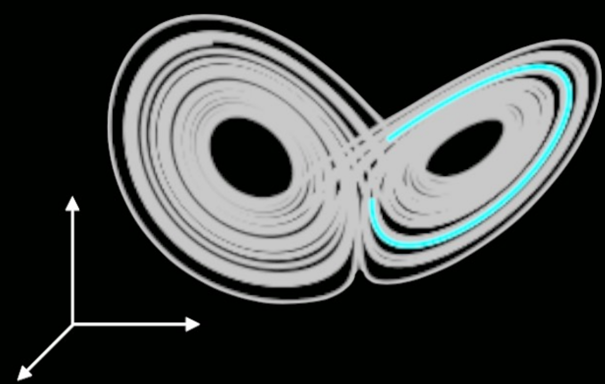
Multiscale Systems

What is this?

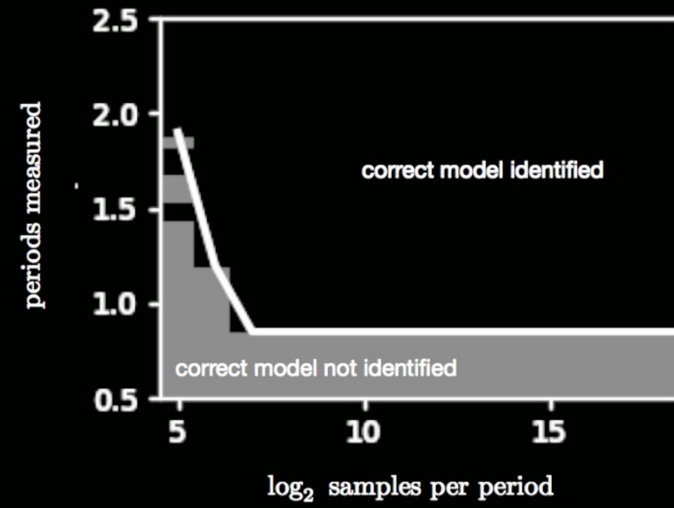

40
30
20
10
0
-10
-20
-30
15.75
15.8
15.85
15.9
15.95
16
16.05
16.1
16.15
16.2
16.25

Limits of Model Discovery

Lorenz system



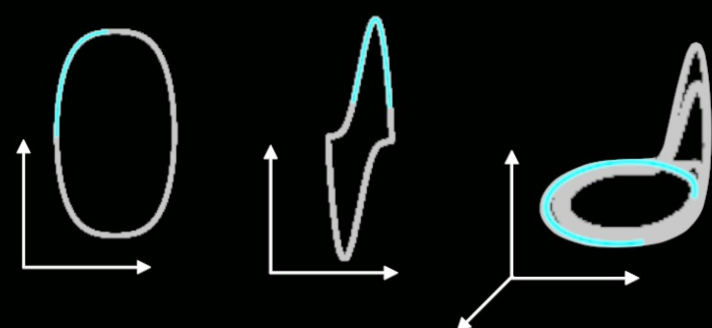
SINDy sampling requirements

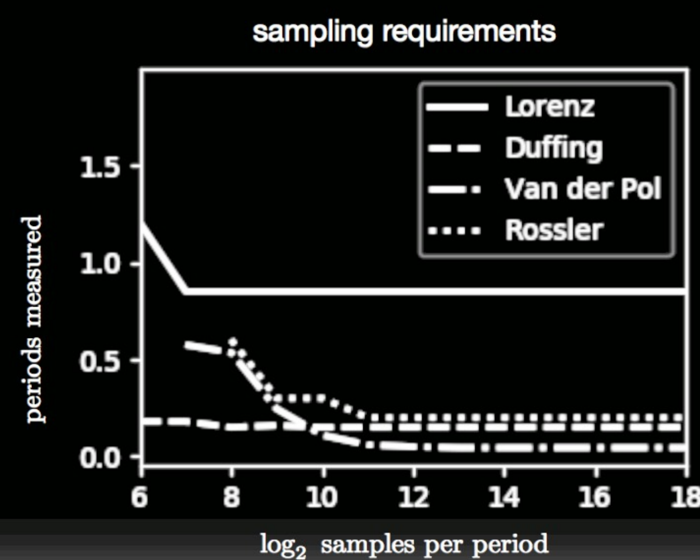


Duffing

Van der Pol

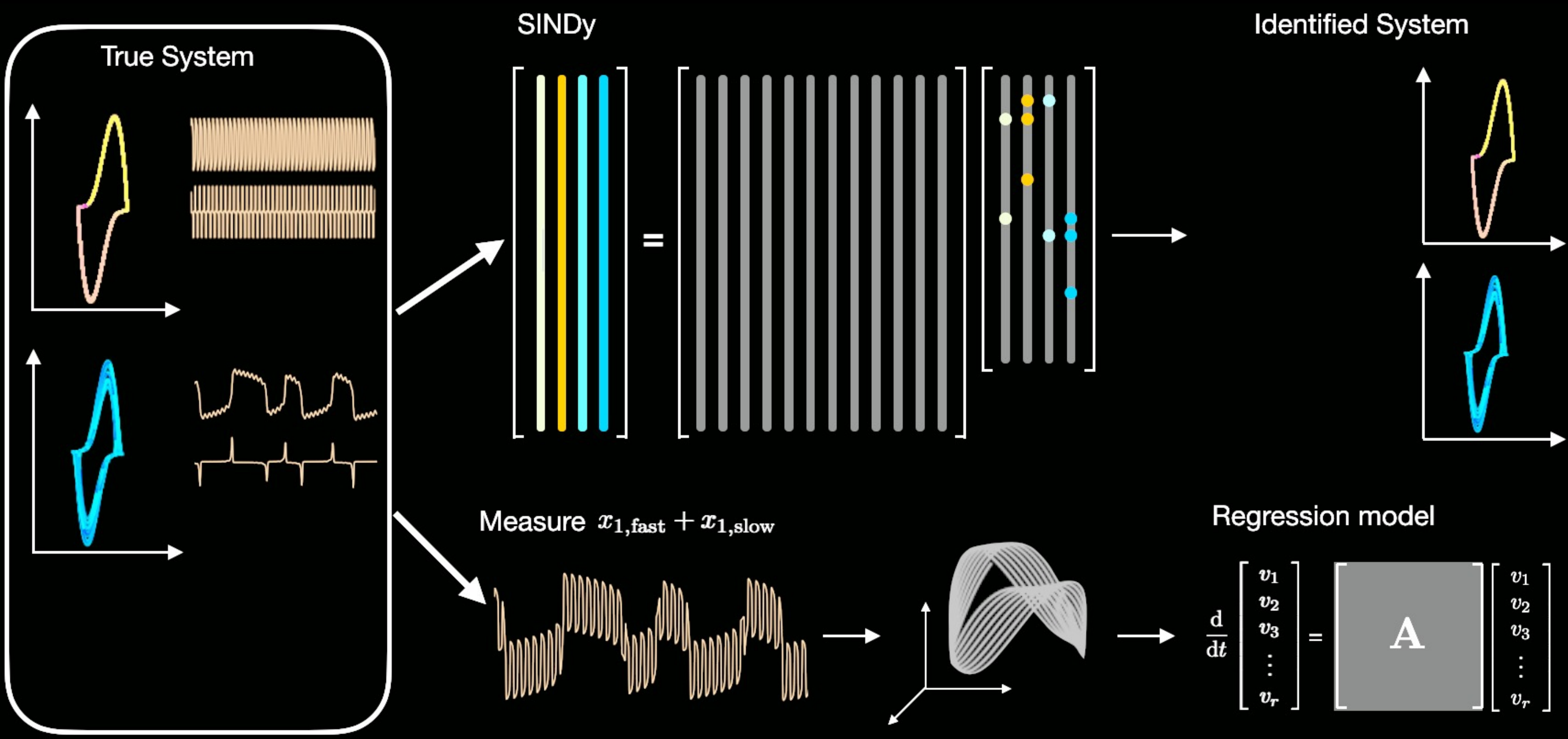
Rossler



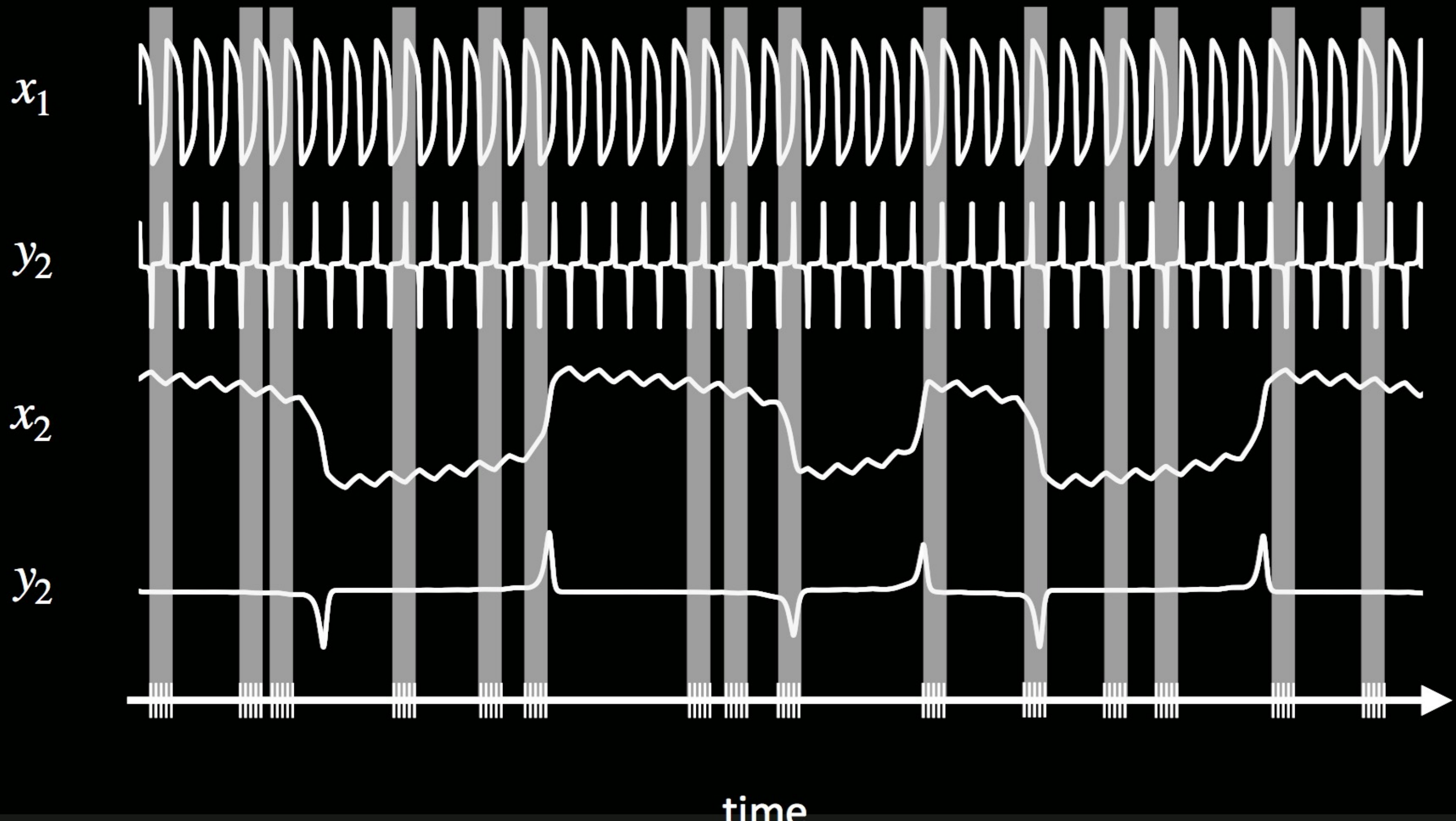


Kathleen Champion

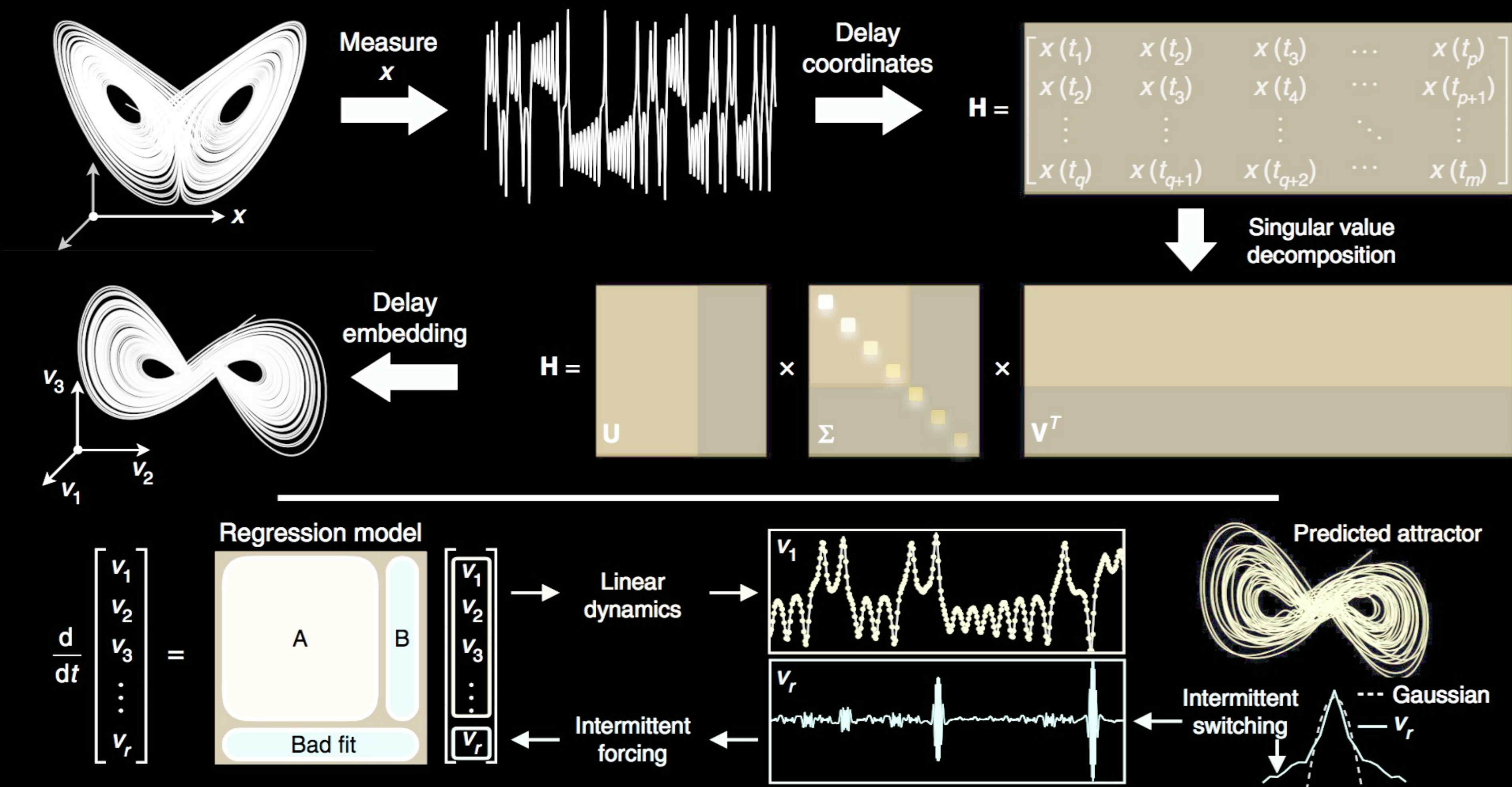
Multiscale Physics Discovery



Burst sampling



Latent Variables

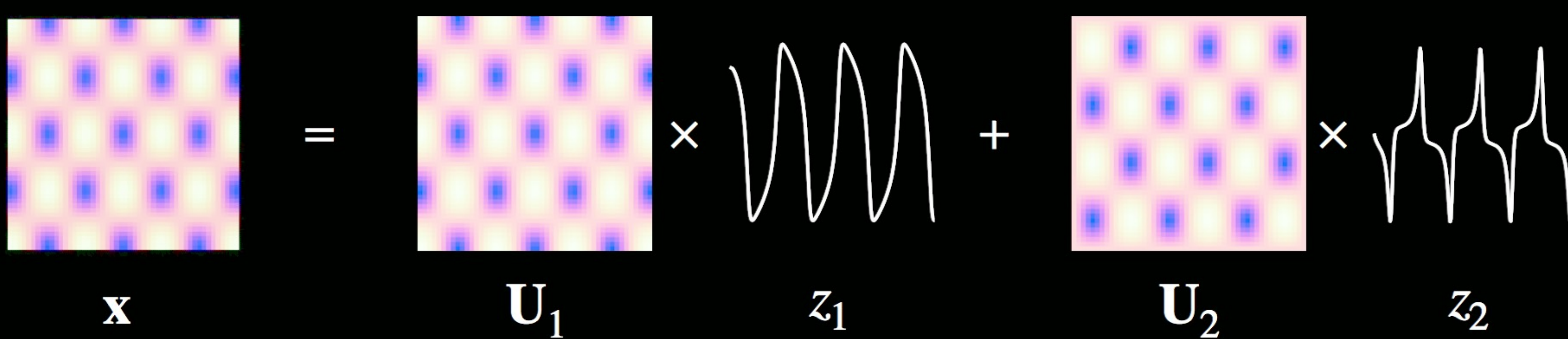


Measure x
x
Delay coordinates
H =
x(t1) x(t2) x(t3) ⋯ x(tp)
x(t2) x(t3) x(t4) ⋯ x(tp+1)
x(tq) x(tq+1) x(tq+2) ⋯ x(tm)
Singular value decomposition
Delay embedding
v3
v2
v1
H =
U
×
Σ
×
V^T
Regression model
d/dt
v1 v2 v3 ⋮ vr
=
A
B
Bad fit
v1 v2 v3 ⋮ vr
Linear dynamics
Intermittent forcing
v1
vr
Predicted attractor
Intermittent switching
Gaussian
vr

Latent variables

$$\mathbf{x} = \mathbf{U}_1 z_1 + \mathbf{U}_2 z_2$$

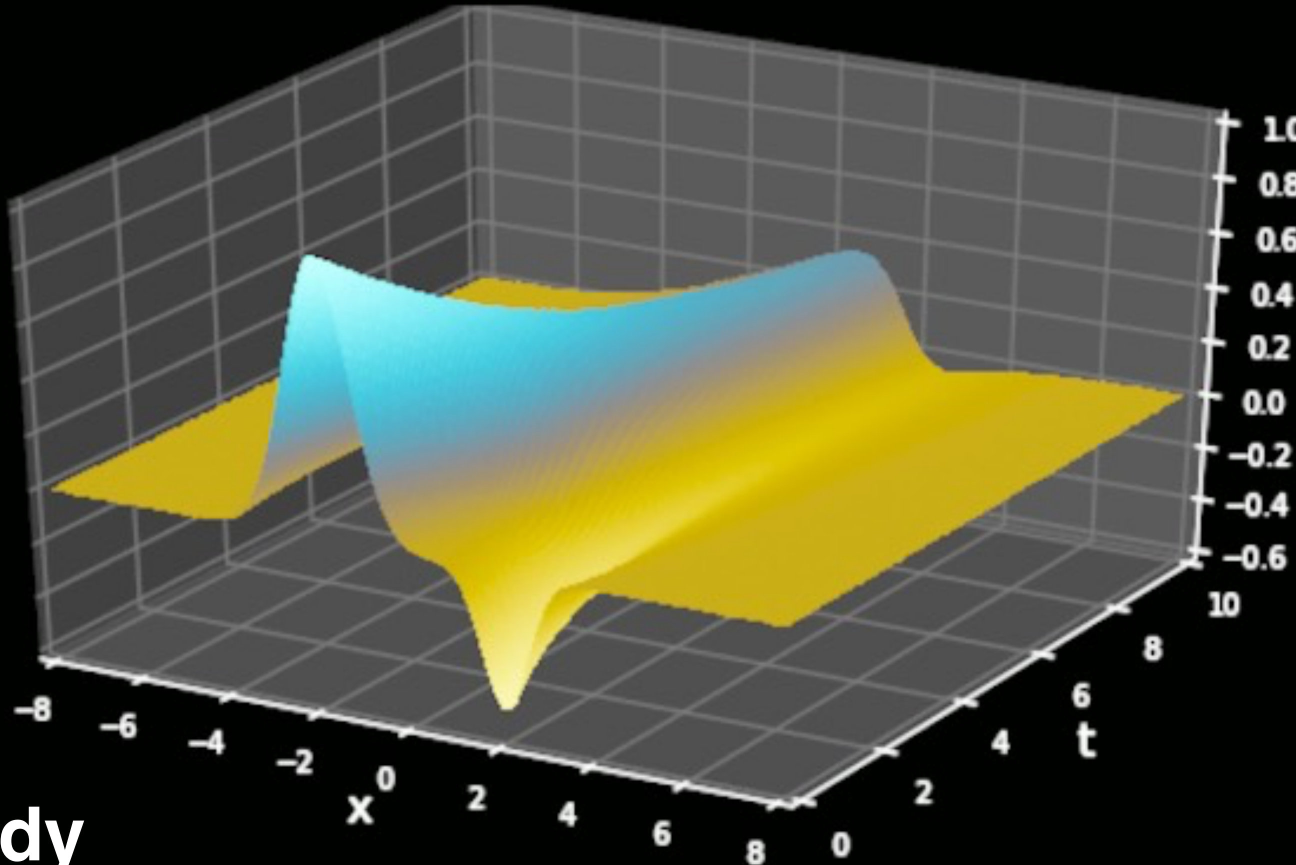
$$\dot{z}_1 = z_2$$
$$\dot{z}_2 = \mu(1 - z_1^2)z_2 - z_1$$



Parametric Systems

Parametric Burgers

$$u_t + \left(1 + \frac{1}{4}\sin(t)\right) u u_x - D u_{xx} = 0$$

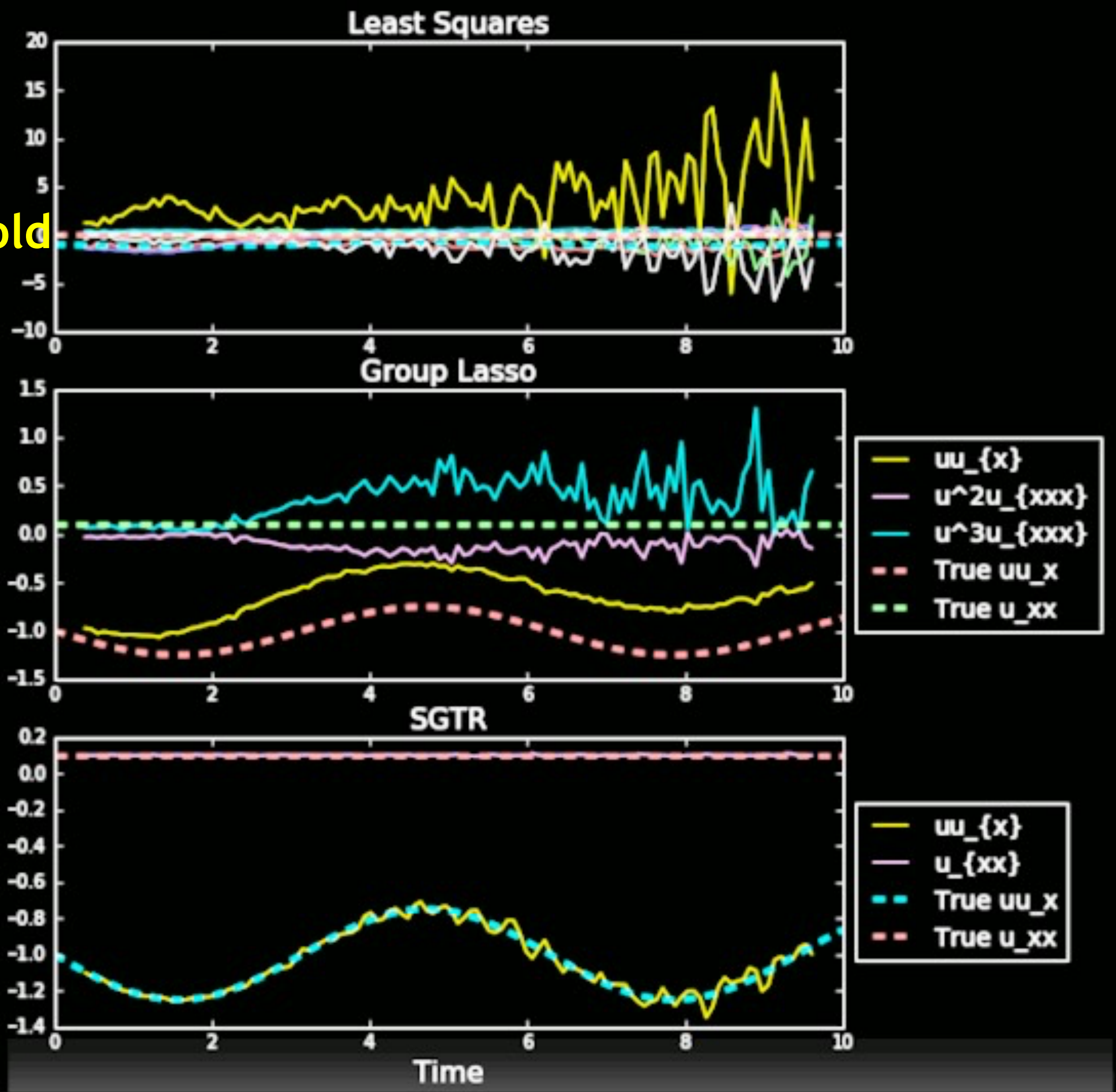


Sam Rudy

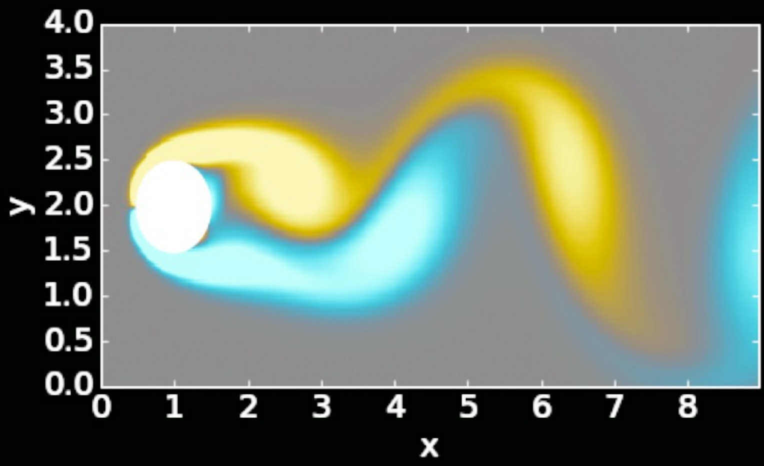
Parametric Discovery

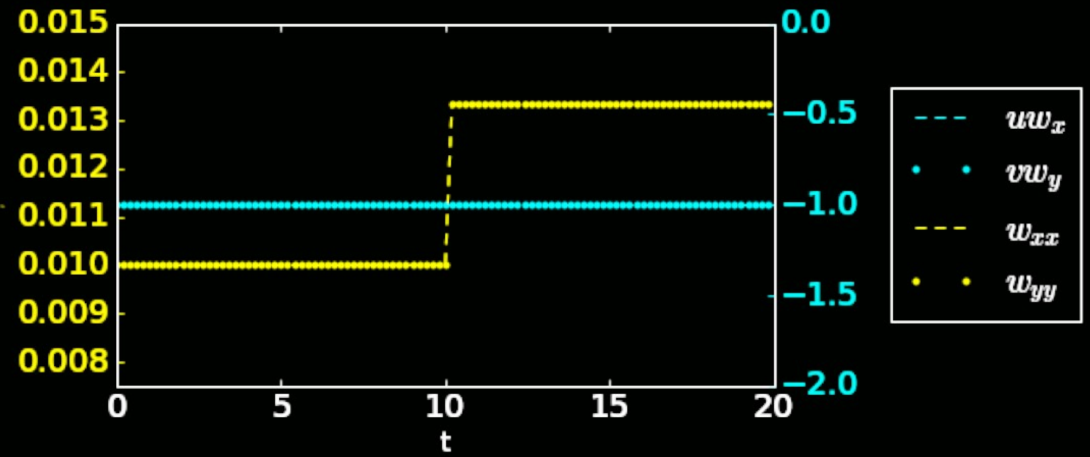
Group LASSO vs
Sequential Group Threshold
Regression (SGTR)

Our innovation: SGTR
(works amazingly well!)

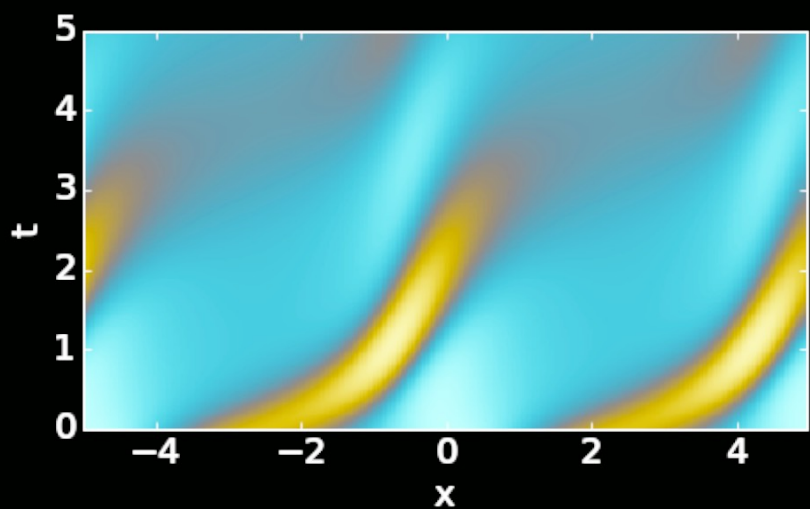


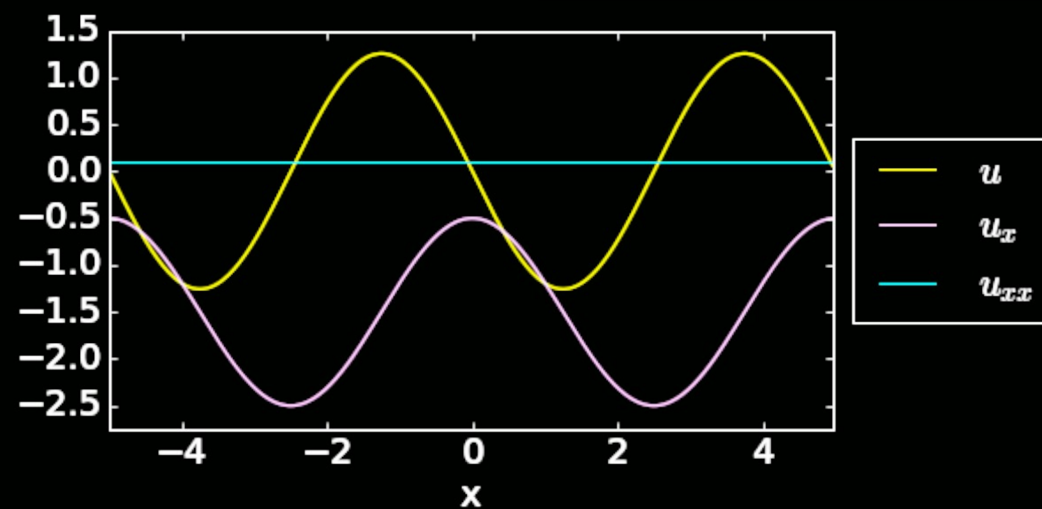
Parametric Dependence



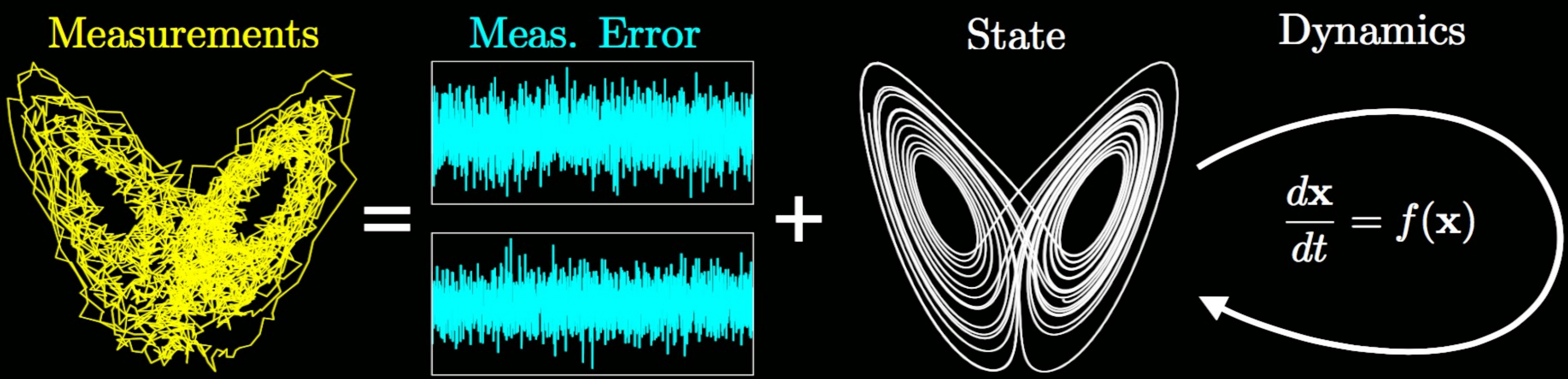


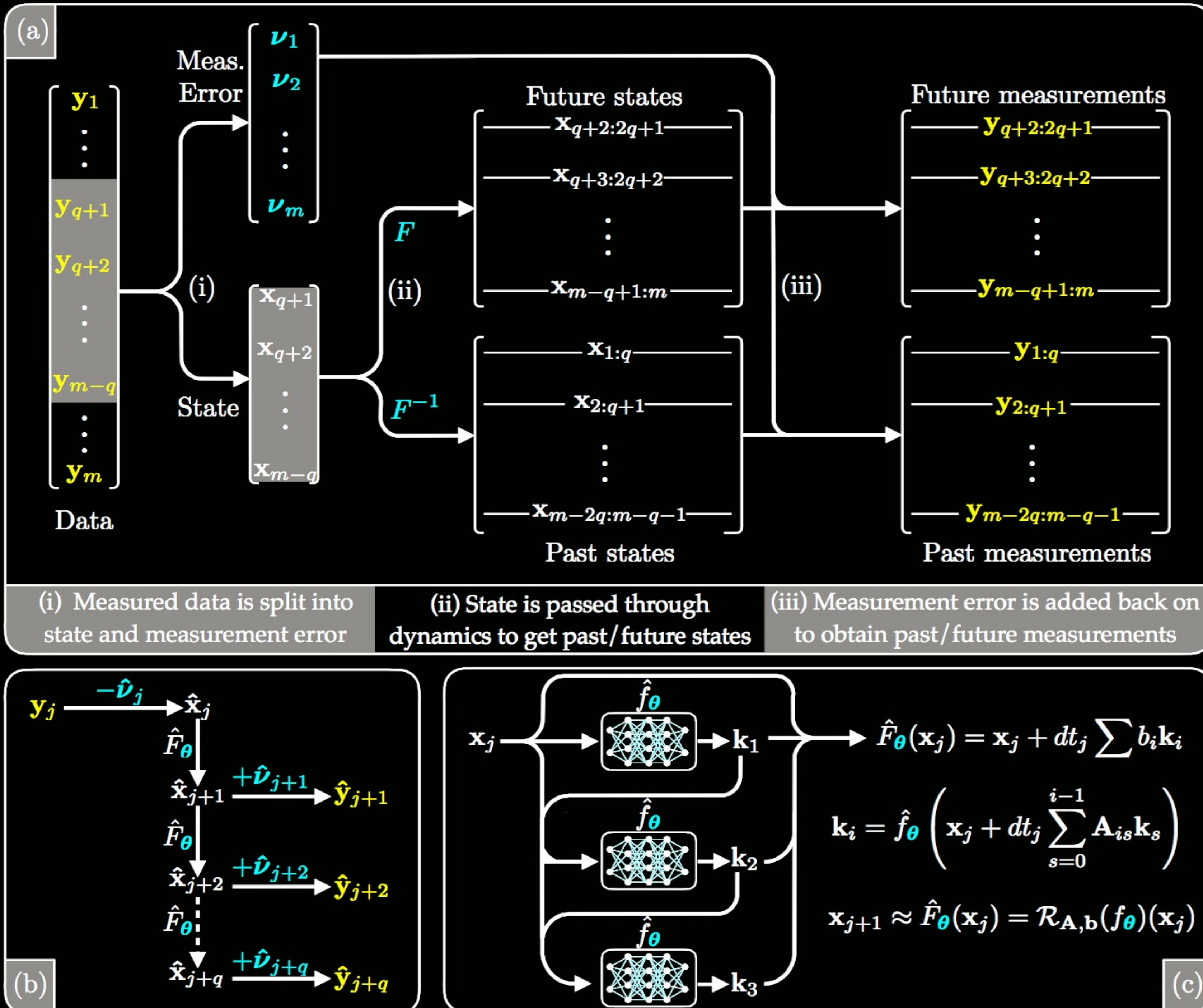
$$u_t = (c(x)u)_x + \epsilon u_{xx}$$

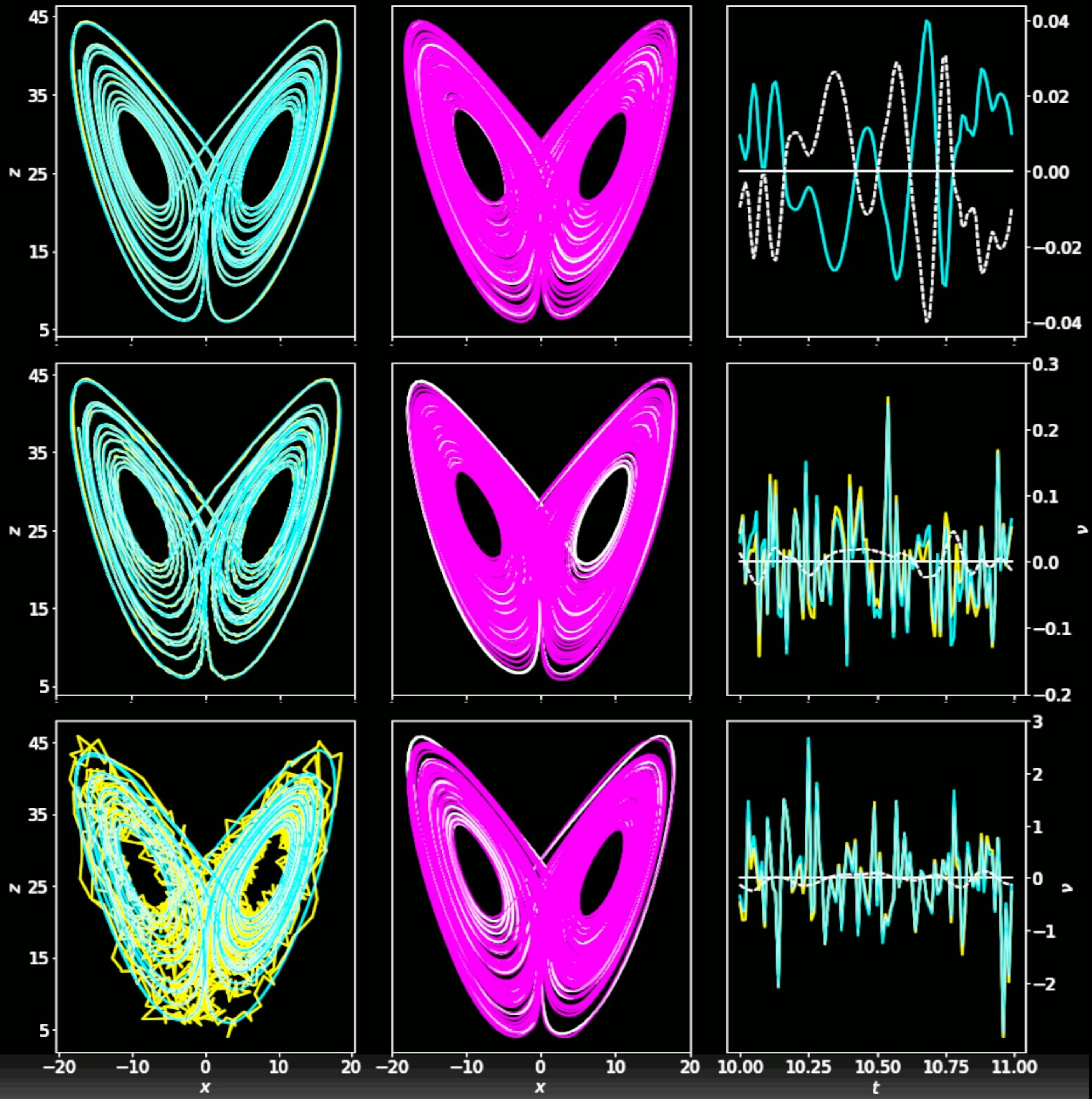




Noise


Measurements
=
Meas. Error
+
State
Dynamics
$\frac{d\mathbf{x}}{dt} = f(\mathbf{x})$


(a)
Data
$\mathbf{y}_1$
$\mathbf{y}_{q+1}$
$\mathbf{y}_{q+2}$
$\mathbf{y}_{m-q}$
$\mathbf{y}_m$
Meas. Error
$\boldsymbol{\nu}_1$
$\boldsymbol{\nu}_2$
$\boldsymbol{\nu}_m$
State
$\mathbf{x}_{q+1}$
$\mathbf{x}_{q+2}$
$\mathbf{x}_{m-q}$
(i)
(ii)
(iii)
F
F^{-1}
Future states
$\mathbf{x}_{q+2:2q+1}$
$\mathbf{x}_{q+3:2q+2}$
$\mathbf{x}_{m-q+1:m}$
Past states
$\mathbf{x}_{1:q}$
$\mathbf{x}_{2:q+1}$
$\mathbf{x}_{m-2q:m-q-1}$
Future measurements
$\mathbf{y}_{q+2:2q+1}$
$\mathbf{y}_{q+3:2q+2}$
$\mathbf{y}_{m-q+1:m}$
Past measurements
$\mathbf{y}_{1:q}$
$\mathbf{y}_{2:q+1}$
$\mathbf{y}_{m-2q:m-q-1}$
(i) Measured data is split into state and measurement error
(ii) State is passed through dynamics to get past/future states
(iii) Measurement error is added back on to obtain past/future measurements
(b)
$\mathbf{y}_j$
$-\hat{\boldsymbol{\nu}}_j$
$\hat{\mathbf{x}}_j$
$\hat{F}_{\boldsymbol{\theta}}$
$\hat{\mathbf{x}}_{j+1}$
$+\hat{\boldsymbol{\nu}}_{j+1}$
$\hat{\mathbf{y}}_{j+1}$
$\hat{\mathbf{x}}_{j+2}$
$+\hat{\boldsymbol{\nu}}_{j+2}$
$\hat{\mathbf{y}}_{j+2}$
$\hat{\mathbf{x}}_{j+q}$
$+\hat{\boldsymbol{\nu}}_{j+q}$
$\hat{\mathbf{y}}_{j+q}$
(c)
$\mathbf{x}_j$
$\hat{f}_{\boldsymbol{\theta}}$
$\mathbf{k}_1$
$\mathbf{k}_2$
$\mathbf{k}_3$
$\hat{F}_{\boldsymbol{\theta}}(\mathbf{x}_j) = \mathbf{x}_j + dt_j \sum b_i \mathbf{k}_i$
$\mathbf{k}_i = \hat{f}_{\boldsymbol{\theta}}\left(\mathbf{x}_j + dt_j \sum_{s=0}^{i-1} \mathbf{A}_{is}\mathbf{k}_s\right)$
$\mathbf{x}_{j+1} \approx \hat{F}_{\boldsymbol{\theta}}(\mathbf{x}_j) = \mathcal{R}_{\mathbf{A},\mathbf{b}}(f_{\boldsymbol{\theta}})(\mathbf{x}_j)$



z
x
t
v
45
35
25
15
5
−20
−10
0
10
20
10.00
10.25
10.50
10.75
11.00
0.04
0.02
0.00
−0.02
−0.04
0.3
0.2
0.1
0.0
−0.1
−0.2
3
2
1
−1
−2

SINDy Innovations

Schaeffer -- corrupt data, PDEs, integral formulation, algorithm convergence
Dongbin Xiu & co-workers (2018) – Sampling strategies
Guang Lin & co-workers (2018) -- Uncertainty Metrics

Zheng, Askham, Brunton, Kutz & Aravkin (2018) – SR3 sparse relaxed regularized regression (for SINDy, LASSO, CS, TV, Matrix Completion …)

Manifolds and Embeddings

Observables & Coordinates

Bernard Koopman 1931

Definition: Koopman Operator (Koopman 1931): *For a dynamical system*

$$\frac{d\mathbf{x}}{dt} = \mathbf{N}(\mathbf{x}),$$

where $\mathbf{x} \in \mathbb{R}^n$ *is in a state space* $\mathbf{x} \in \mathcal{M}$. *The Koopman operator* $\mathcal{K}$ *acts on a set of scalar observable variables* g_j *which comprise the vector* $\mathbf{g} : \mathcal{M} \to \mathbb{C}$ *so that*

$$\mathcal{K}\mathbf{g}(\mathbf{x}) = \mathbf{g}(\mathbf{N}(\mathbf{x})) .$$

Dynamic Mode Decomposition

Definition: Dynamic Mode Decomposition (Tu et al. 2014 [1]): *Suppose we have a dynamical system (1.17) and two sets of data*

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_M \\ | & | & & | \end{bmatrix}$$

$$\mathbf{X}' = \begin{bmatrix} | & | & & | \\ \mathbf{x}'_1 & \mathbf{x}'_2 & \cdots & \mathbf{x}'_M \\ | & | & & | \end{bmatrix}$$

with $\mathbf{x}_k$ an initial condition to (1.17) and $\mathbf{x}'_k$ it corresponding output after some prescribed evolution time τ with there being m initial conditions considered. The DMD modes are eigenvectors of

$$\mathrm{A}_{\mathbf{X}} = \mathbf{X}'\mathbf{X}^{\dagger}$$

where $\dagger$ denotes the Moore-Penrose pseudoinverse.

Travis Askham — optimized DMD

Approximate Dynamical Systems

Linear dynamics (equation-free)

$$\frac{d\tilde{\mathbf{x}}}{dt} = \mathbf{A}\tilde{\mathbf{x}}$$

Eigenfunction expansion

$$\tilde{\mathbf{x}}(t) = \sum_{k=1}^{K} b_k \psi_k \exp(\omega_k t)$$

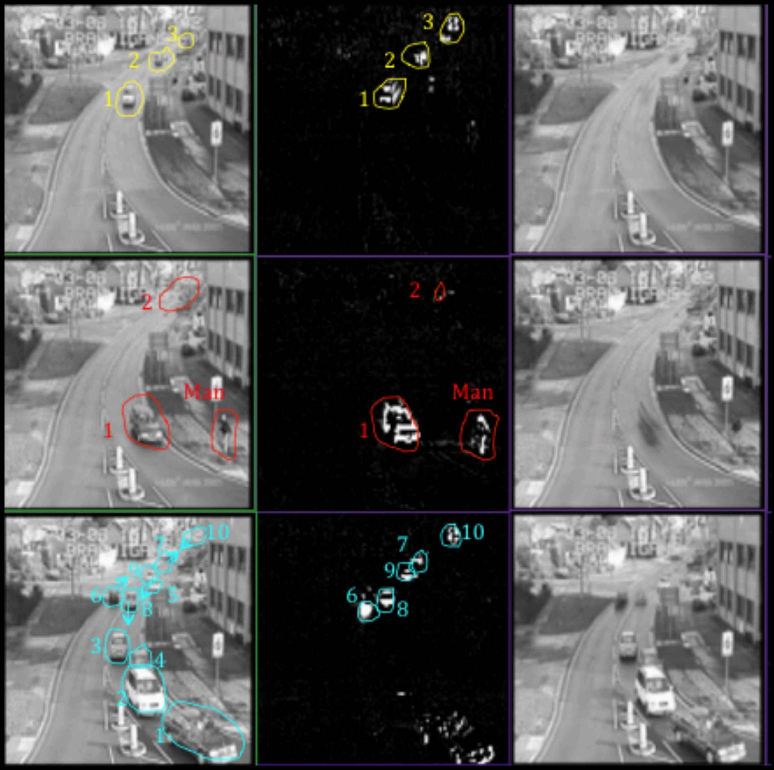
Least-square fit

$$\|\mathbf{x}(t) - \tilde{\mathbf{x}}(t)\| \ll 1$$

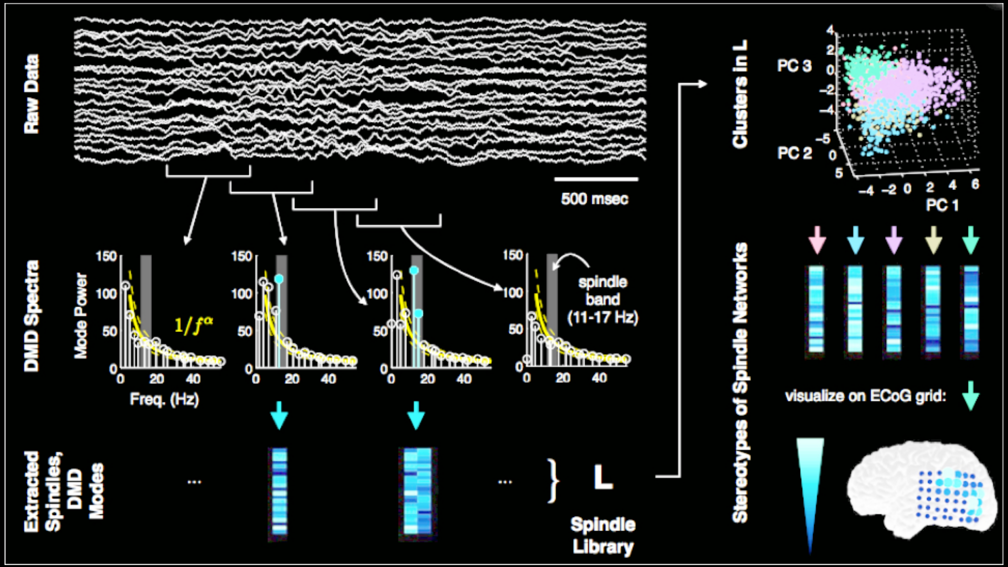
Some Applications

Dynamic Mode Decomposition for Financial Trading Strategies

Jordan Mann* and J. Nathan Kutz‡†



ECOG recordings



Erichson, Brunton & Kutz (2017)
Brunton, Johnson, Ojemann & Kutz (2017)

DMD with Control

Input

$$\mathbf{x}_{k+1} \approx \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$$

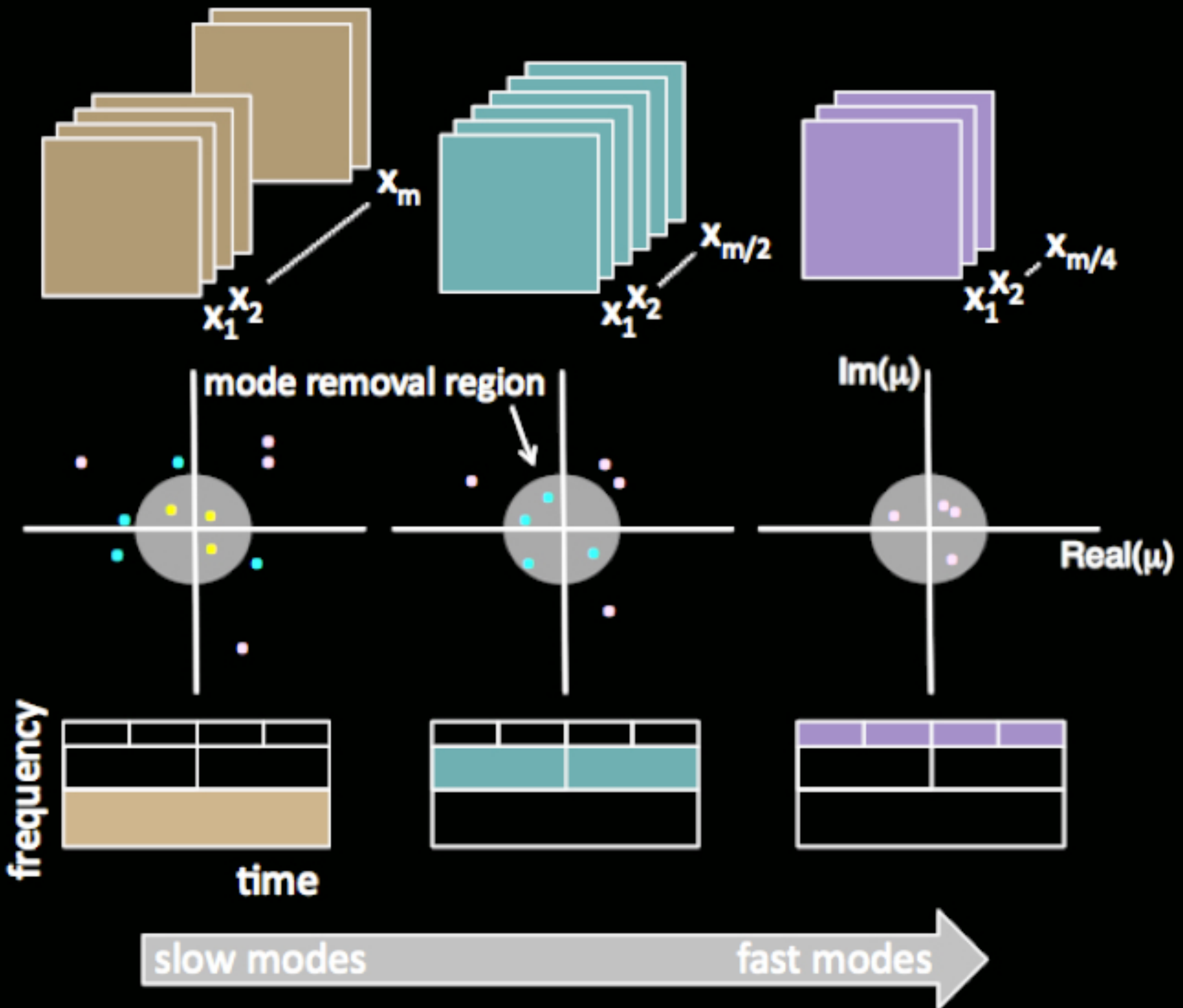
Input Snapshots

$$\mathbf{\Upsilon} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_{m-1} \\ | & | & & | \end{bmatrix}$$

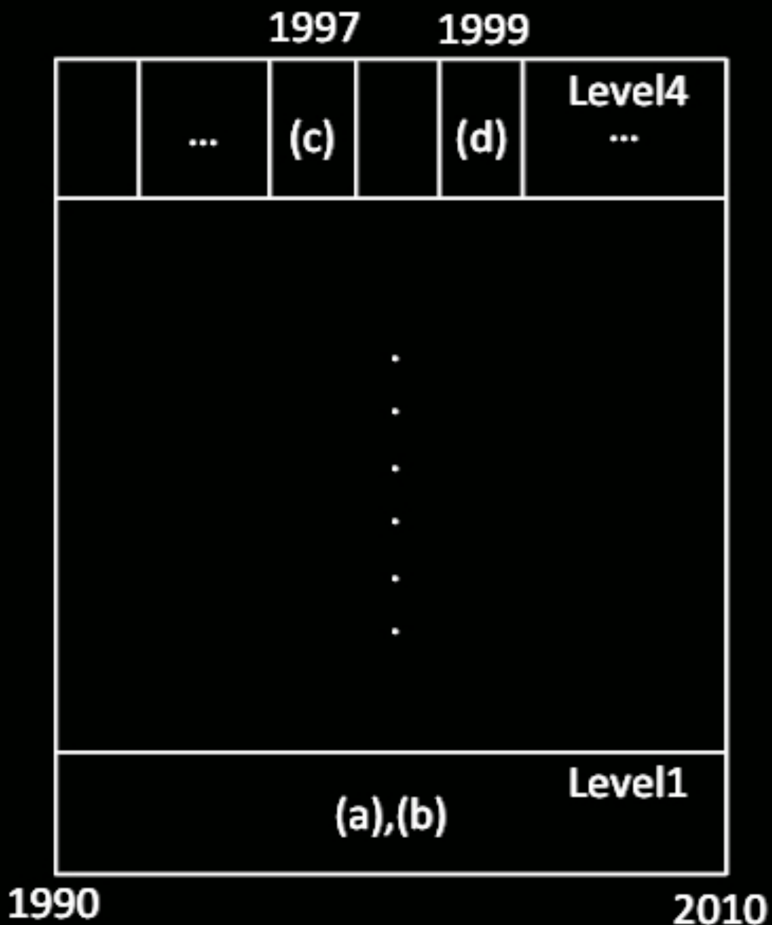
DMD generalization

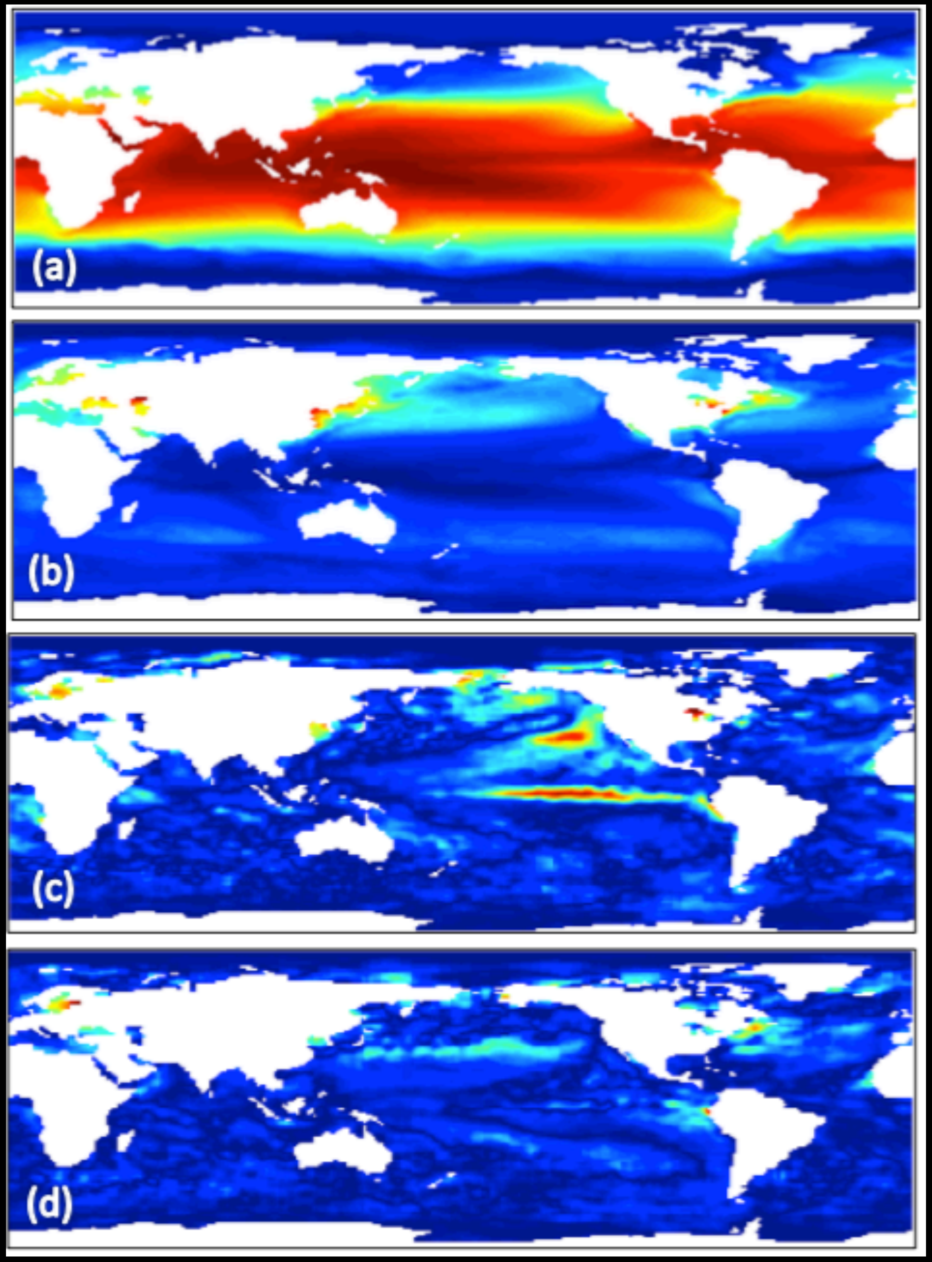
$$\mathbf{X}' \approx \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{\Upsilon}$$

Multi-Resolution DMD

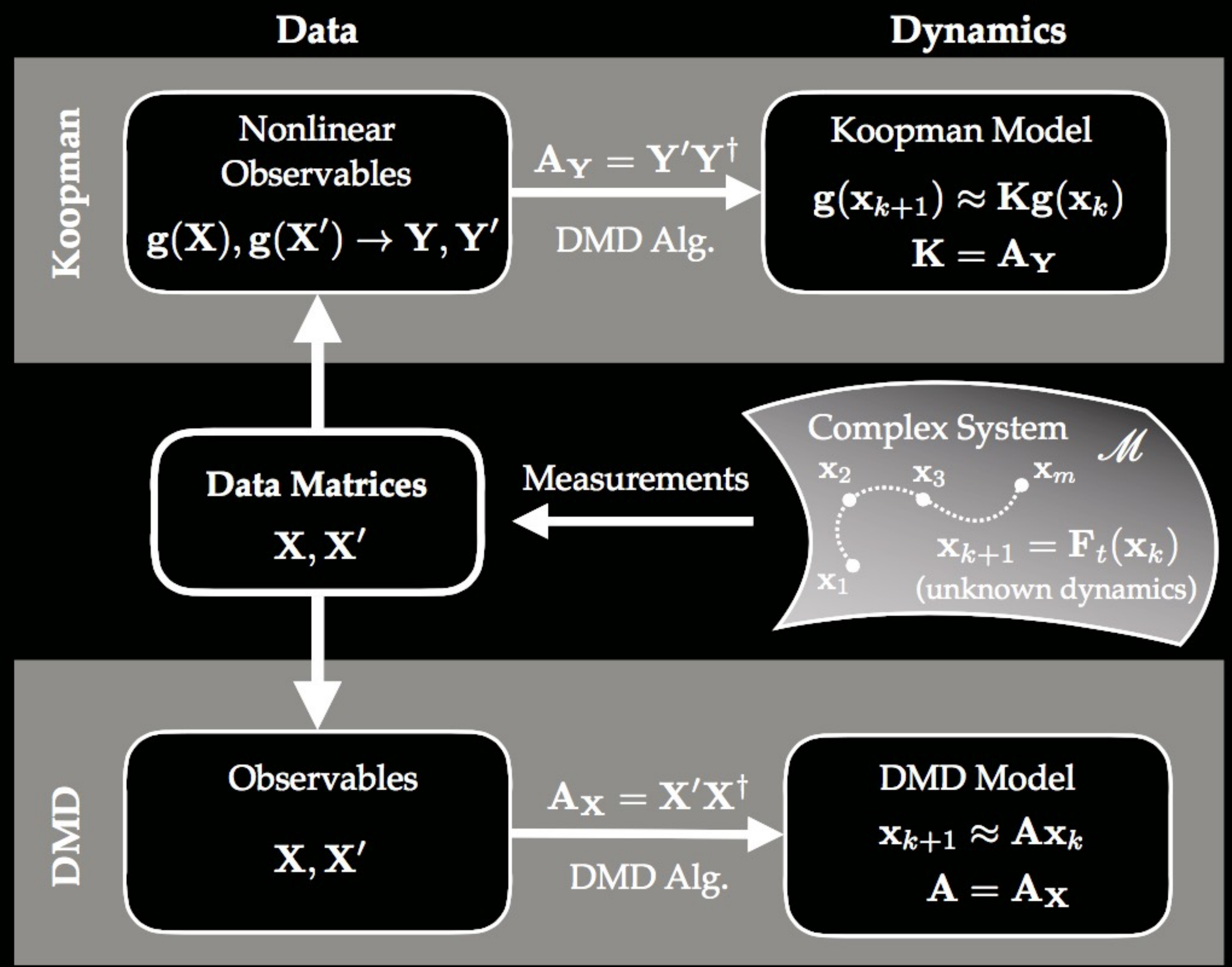


SST data & El Nino (1990s-2010+)



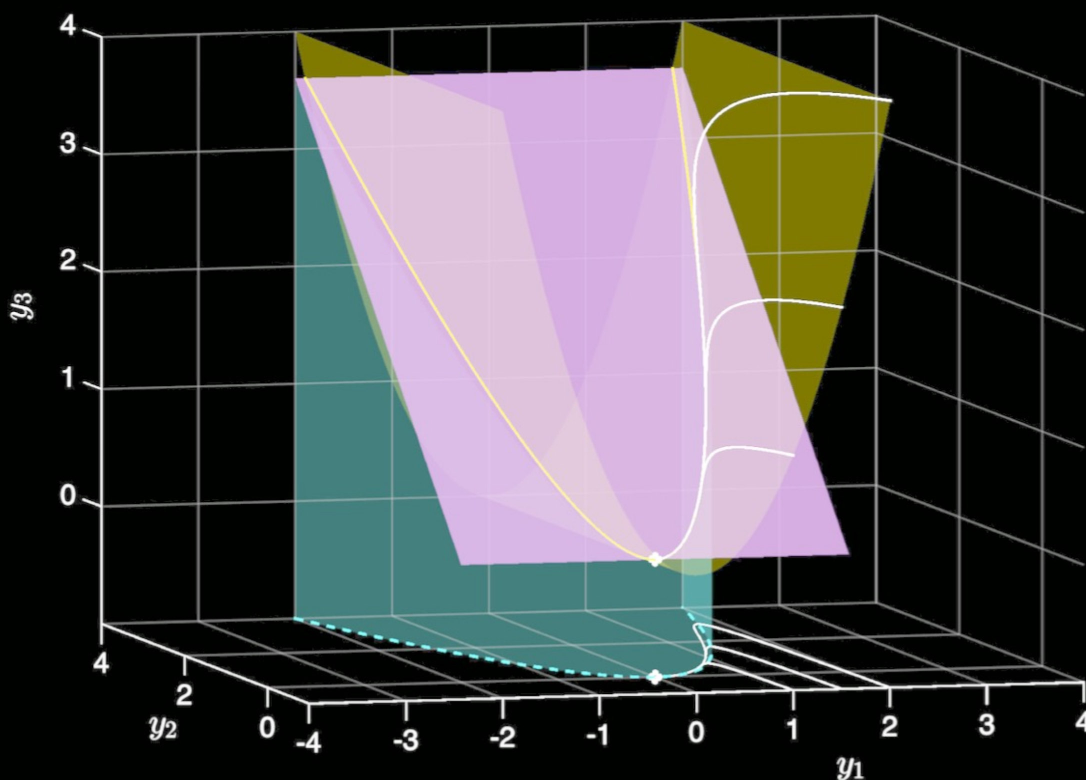


Koopman vs DMD: All about Observables!



Koopman Invariant Subspaces

$$\left.\begin{aligned}\dot{x}_1 &= \mu x_1 \\ \dot{x}_2 &= \lambda(x_2 - x_1^2)\end{aligned}\right\} \implies \frac{d}{dt}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} \mu & 0 & 0 \\ 0 & \lambda & -\lambda \\ 0 & 0 & 2\mu \end{bmatrix}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} \quad \text{for} \quad \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \end{bmatrix}$$



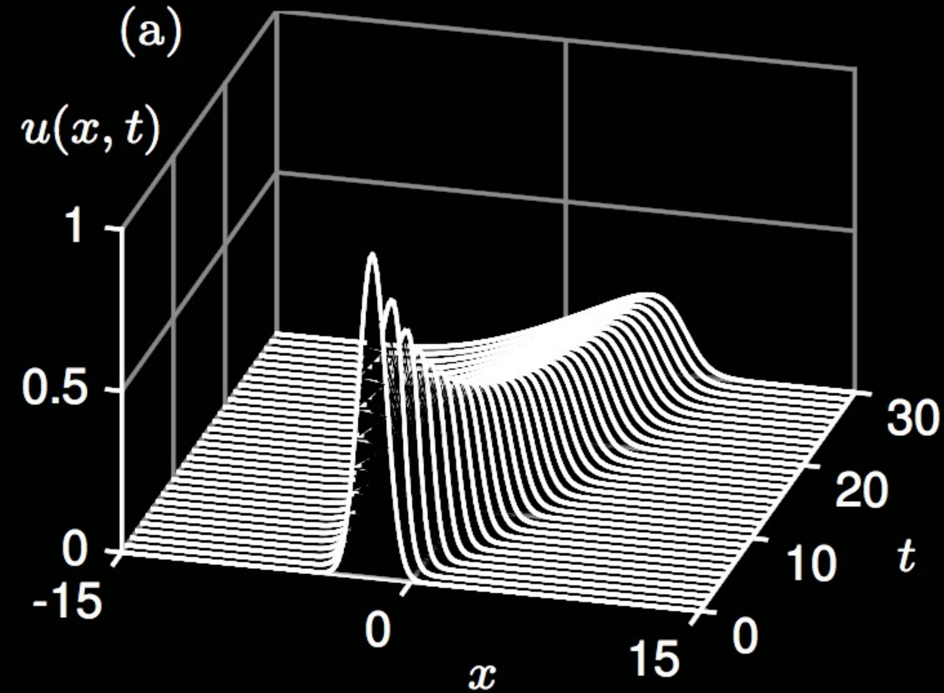
Burgers' Equation

$$u_t + uu_x - \epsilon u_{xx} = 0 \qquad \epsilon > 0, \ x \in [-\infty, \infty]$$

Cole-Hopf

$$u = -2\epsilon v_x / v$$

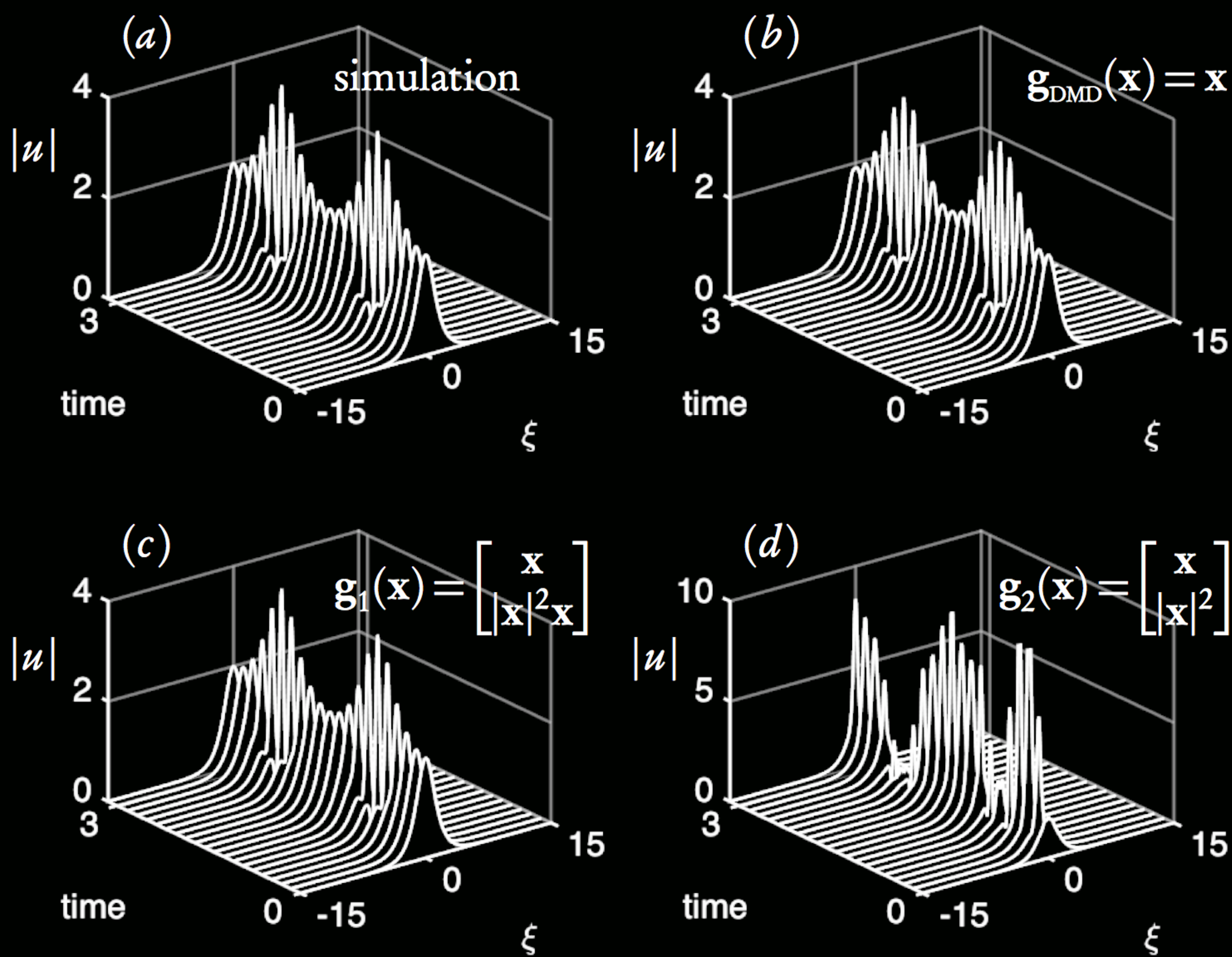
$$v_t = \epsilon v_{xx}$$



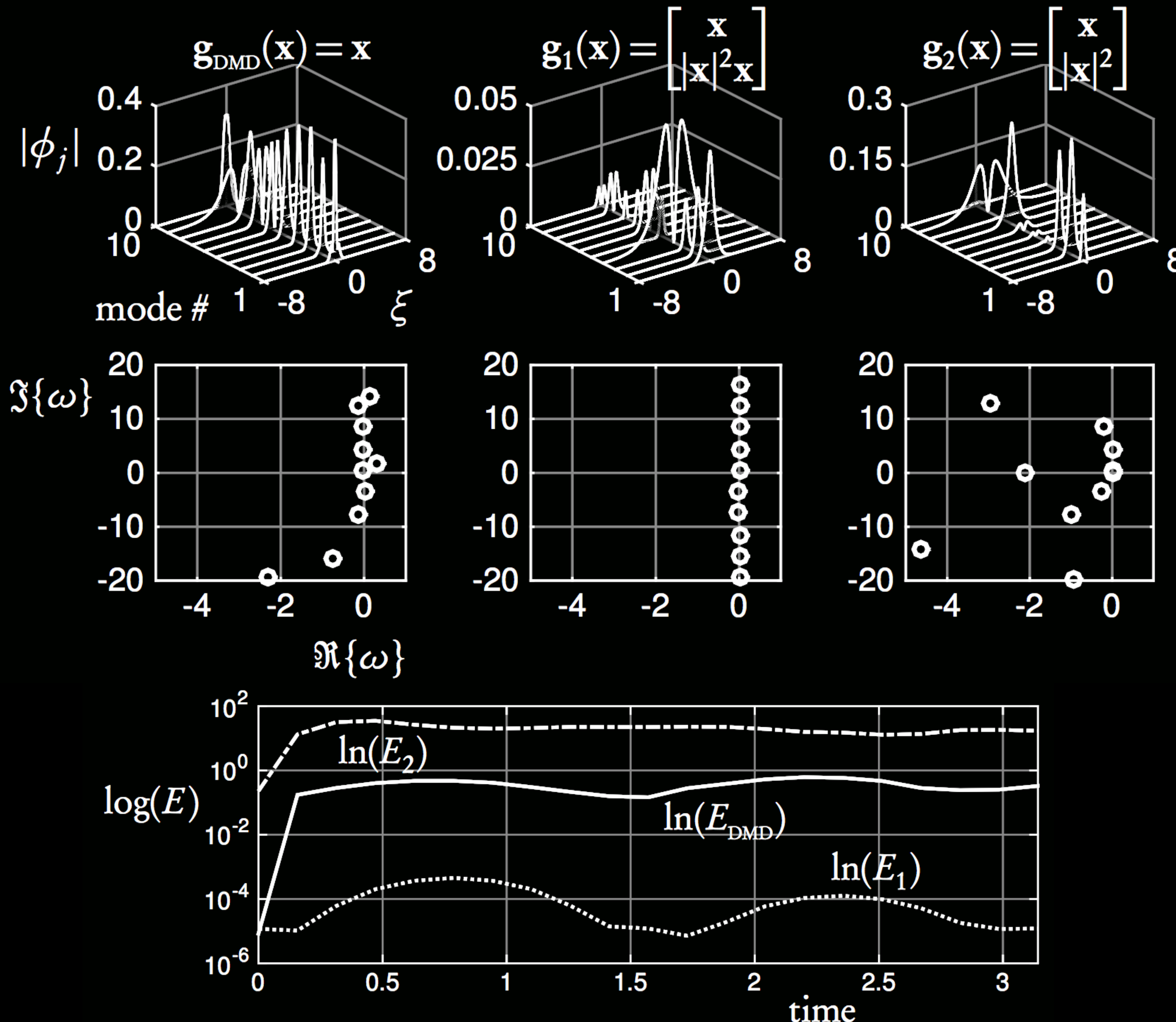
Kutz, Proctor & Brunton, Complexity (2018)

Nonlinear Schrodinger Equation

$$i\frac{\partial u}{\partial t}+\frac{1}{2}\frac{\partial^2 u}{\partial \xi^2}+|u|^2u=0$$



Error and DMD Modes

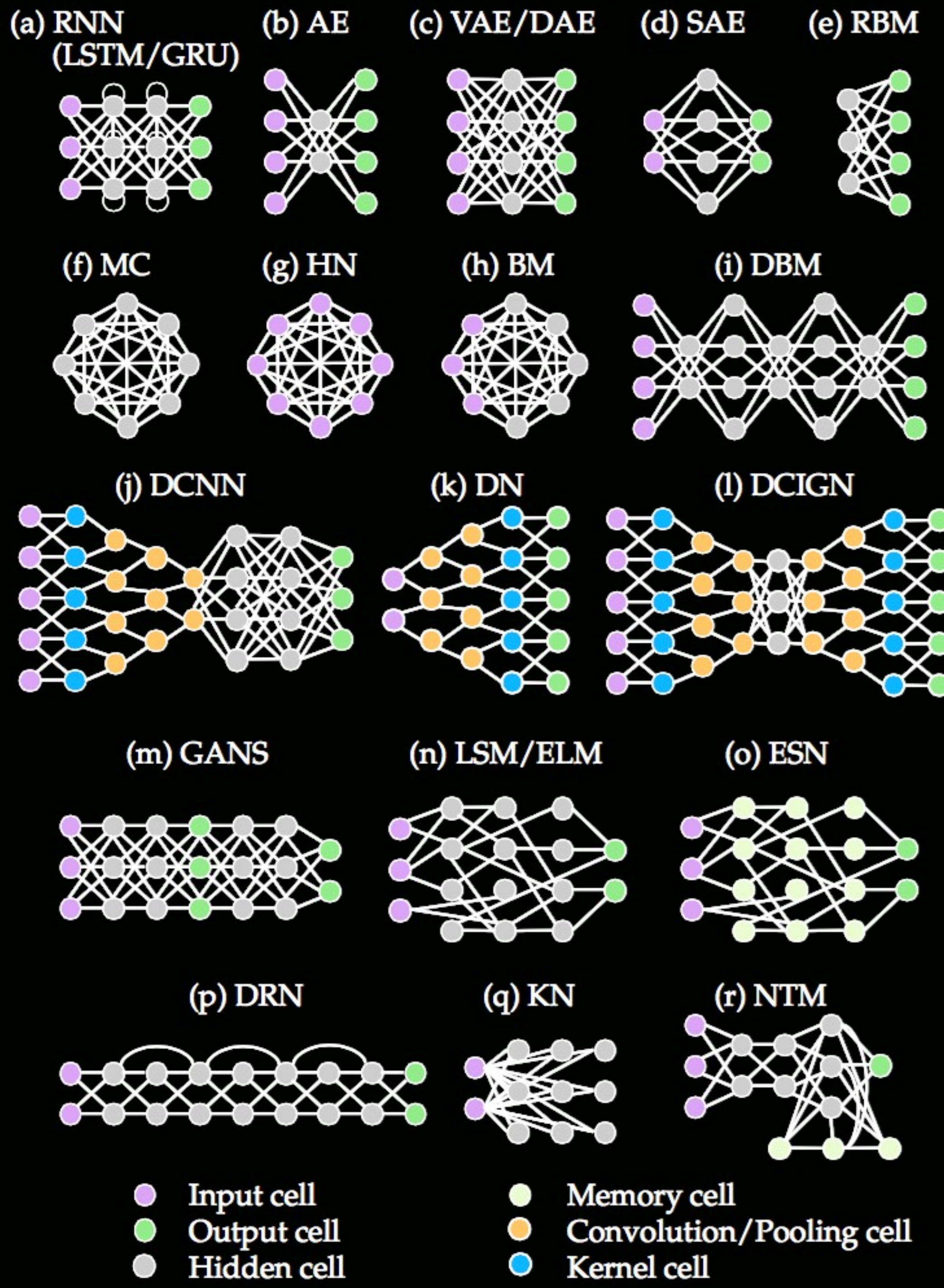


Neural Nets

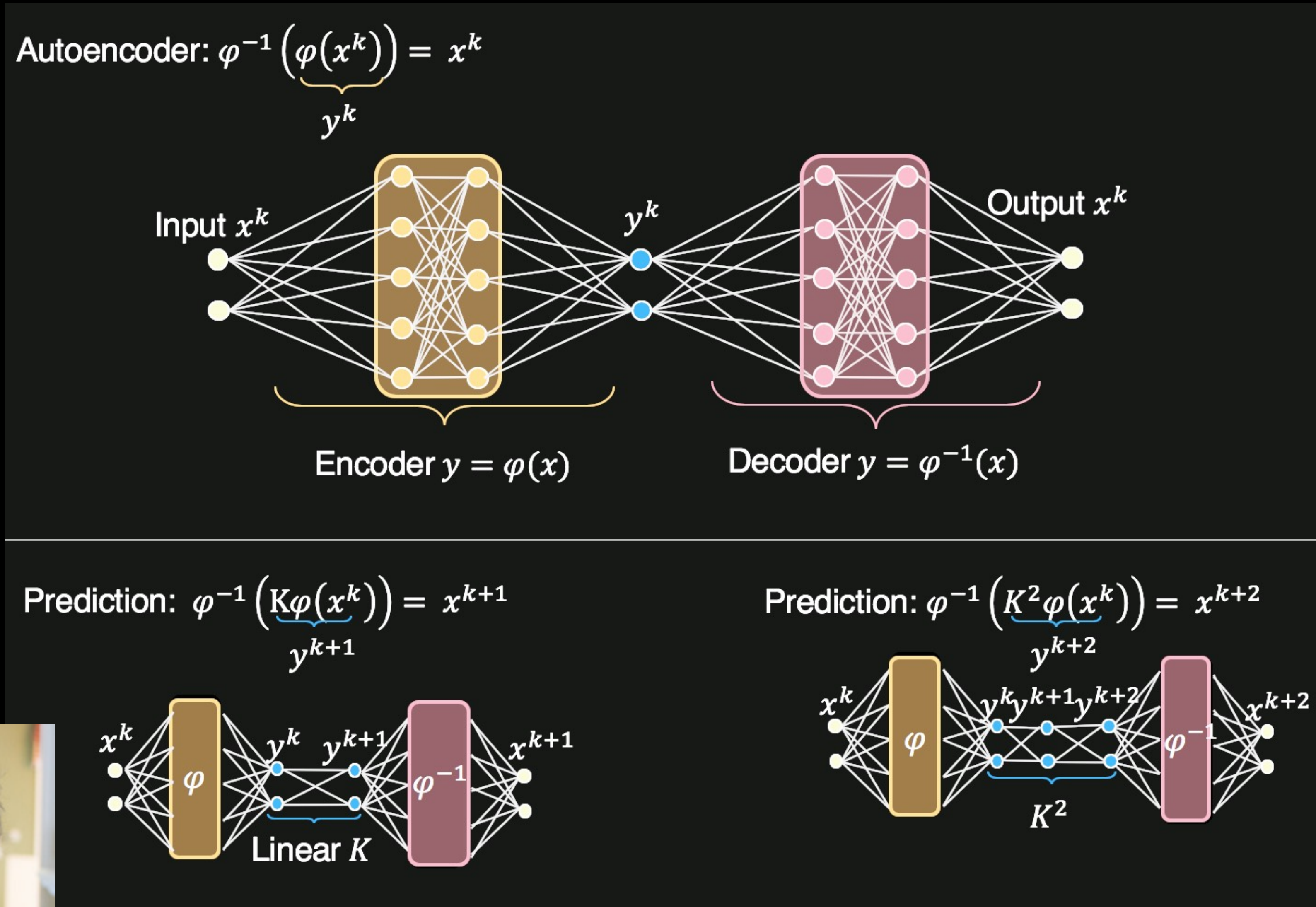
"Supervised learning is a high-dimensional interpolation problem."

S. Mallat, PRSA (2016)

NN Zoo


(a) RNN (LSTM/GRU)
(b) AE
(c) VAE/DAE
(d) SAE
(e) RBM
(f) MC
(g) HN
(h) BM
(i) DBM
(j) DCNN
(k) DN
(l) DCIGN
(m) GANS
(n) LSM/ELM
(o) ESN
(p) DRN
(q) KN
(r) NTM
Input cell
Output cell
Hidden cell
Memory cell
Convolution/Pooling cell
Kernel cell

NNs for Koopman Embedding



Bethany Lusch

Lusch et al. Nat. Comm (2018)

Failure!

(obviously)

Duffing Oscillator

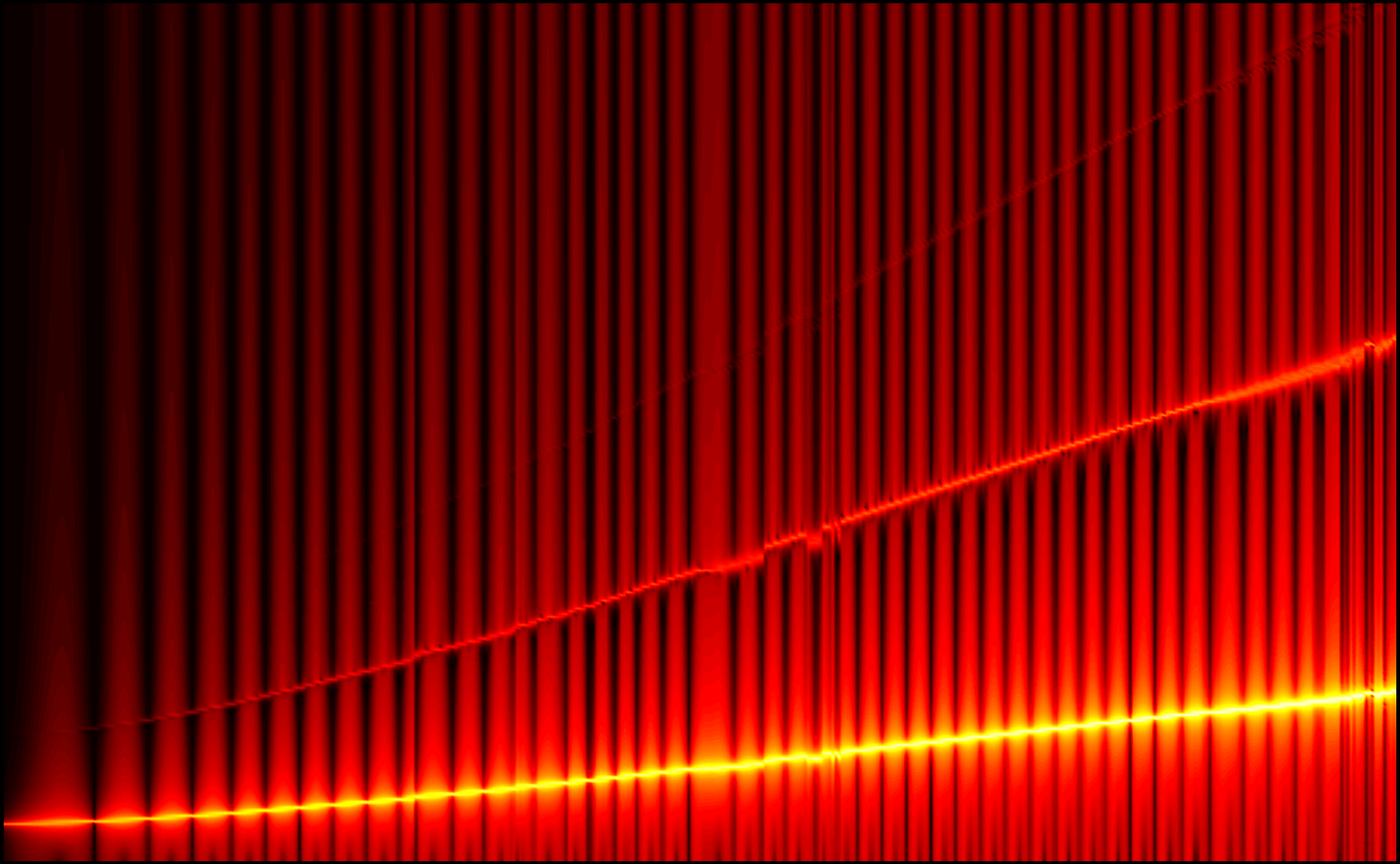
Poincare-Lindstedt Expansion: let $\tau = \omega t$ so that

$$y_{tt} + y + \epsilon y^3 = 0 \Rightarrow \omega^2 y_{\tau\tau} + y + \epsilon y^3 = 0$$

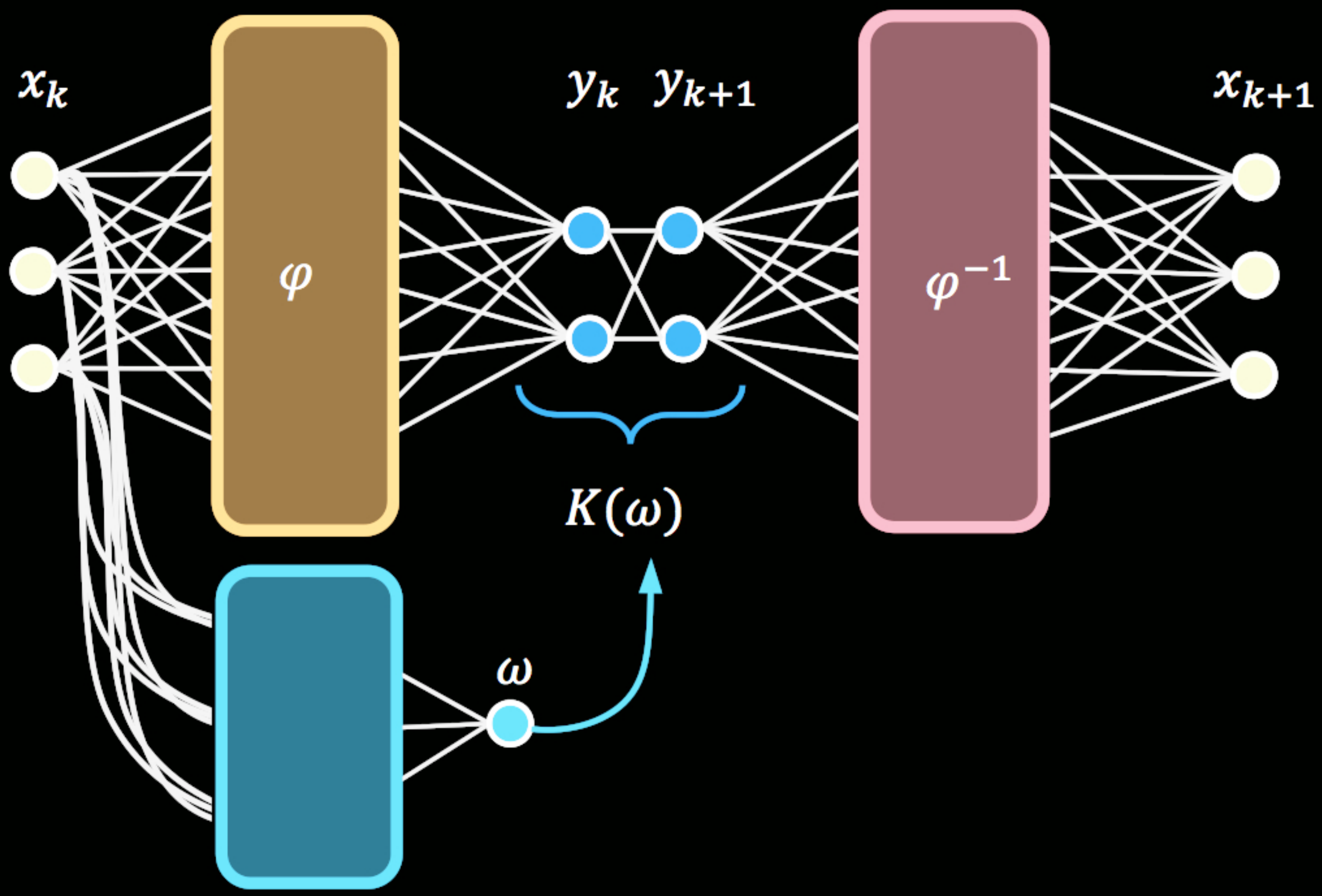
Nonlinearity: Shifts Frequencies + Generates Harmonics

$$y = A \sin[(1+\epsilon 3A^2/8)t] + \epsilon\left\{\frac{3A^3}{32}\sin\left[\left(1+\epsilon\frac{3A^2}{8}\right)t\right] - \frac{A^3}{32}\sin\left[3\left(1+\epsilon\frac{3A^2}{8}\right)t\right]\right\}$$

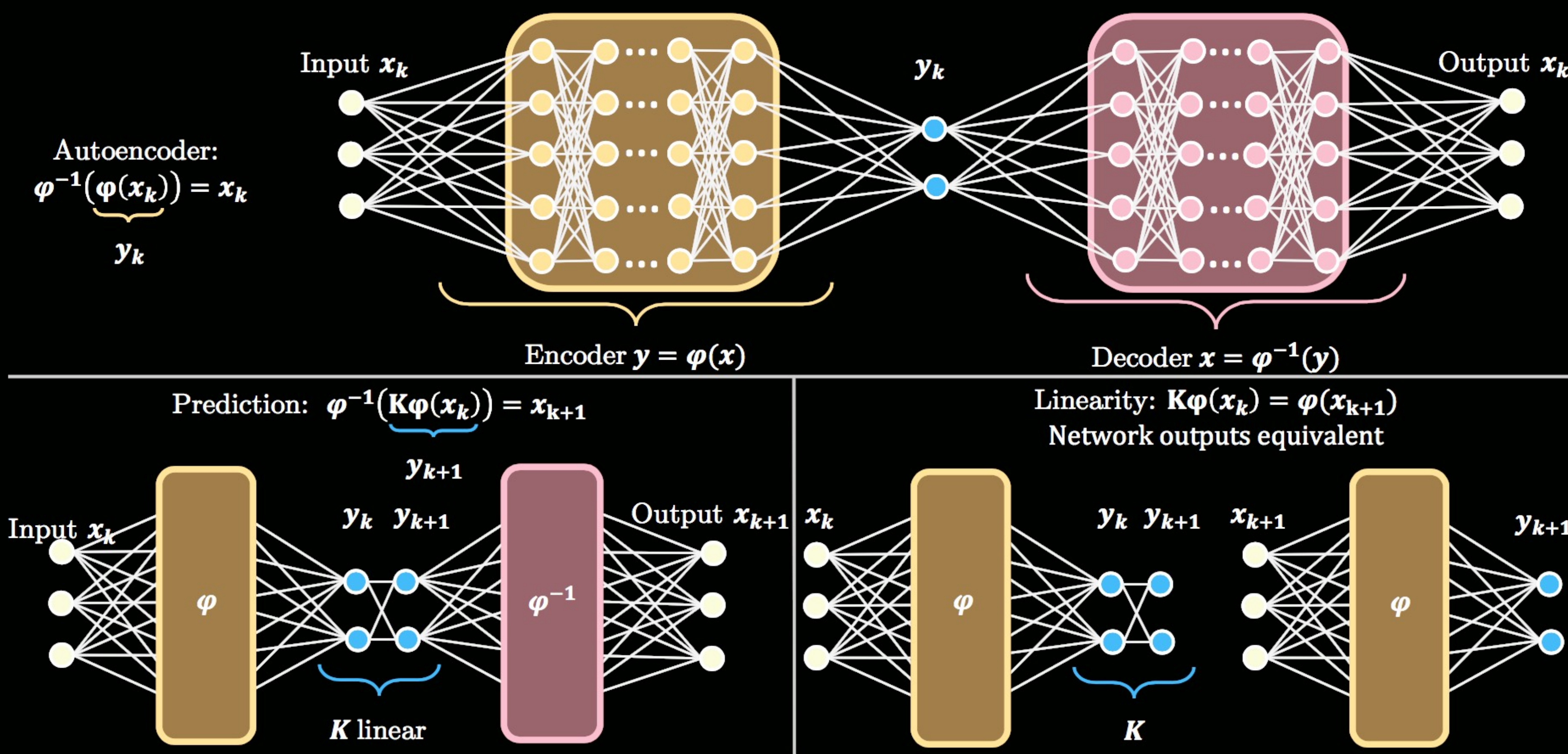
Spectrogram



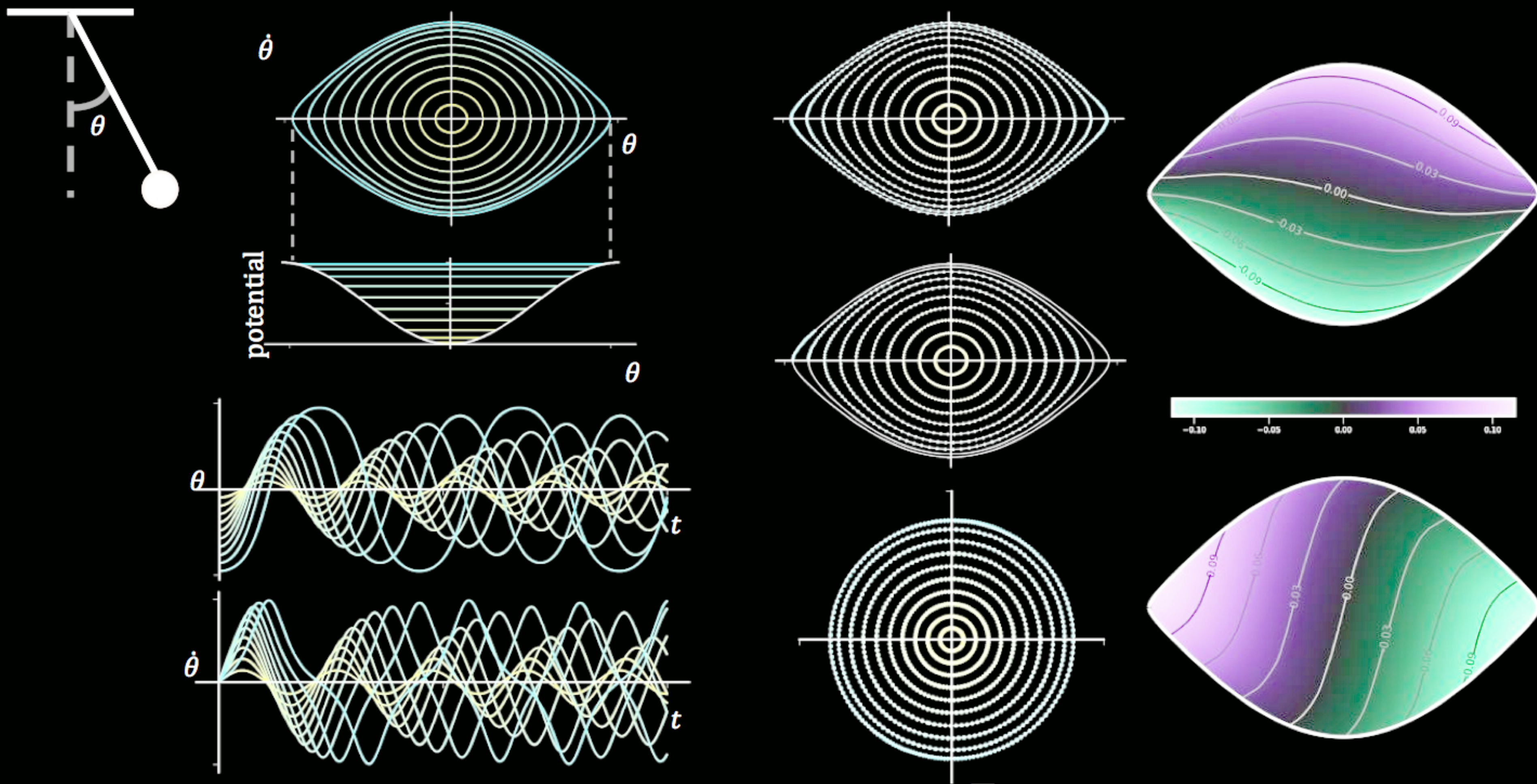
Handling the Continuous Spectra



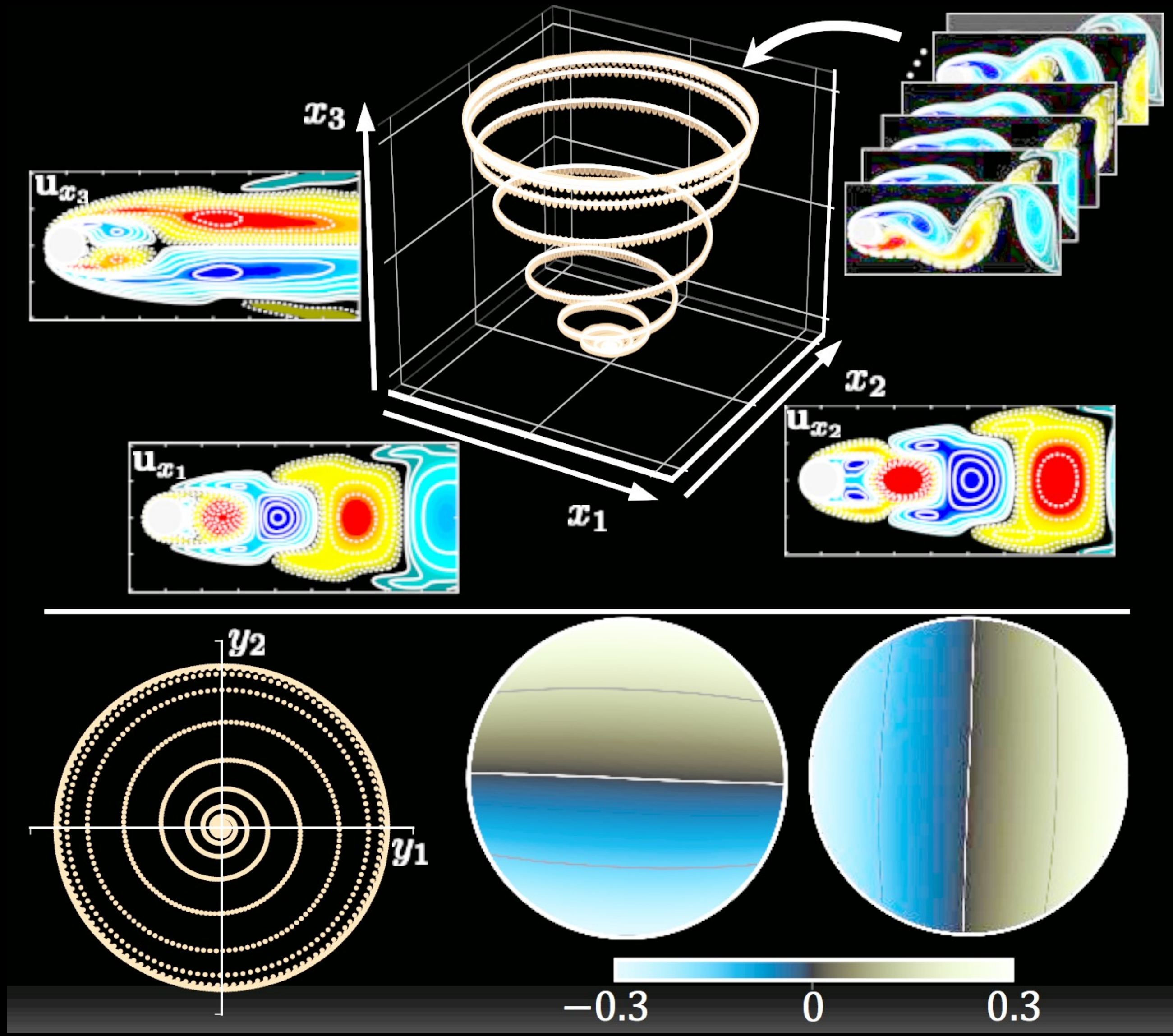
Training Loss Function



The Pendulum



Flow Around a Cylinder



Relax Koopman

Sparse Identification of Nonlinear Dynamics (SINDy)

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t))$$

⬇

$$\mathbf{x}(t) \in \mathbb{R}^n$$

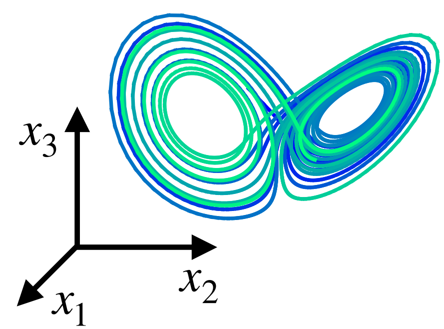
Sparse Identification of Nonlinear Dynamics (SINDy)

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t))$$

$$\mathbf{x}(t) \in \mathbb{R}^n$$

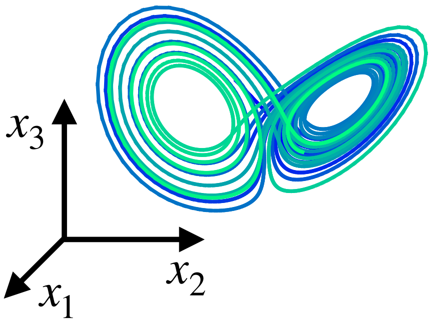
Example: Lorenz

$$\dot{x}_1 = \sigma(x_2 - x_1)$$
$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$
$$\dot{x}_3 = x_1 x_2 - \beta x_3$$



Sparse Identification of Nonlinear Dynamics (SINDy)

True System

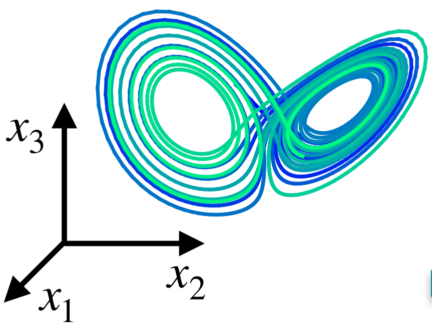


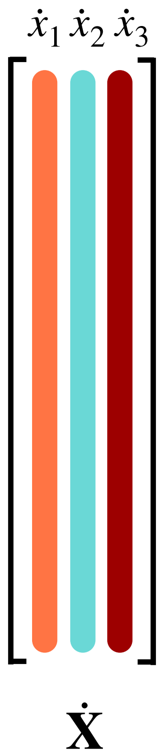
$$\dot{x}_1 = \sigma(x_2 - x_1)$$
$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$
$$\dot{x}_3 = x_1 x_2 - \beta x_3$$

Sparse Identification of Nonlinear Dynamics (SINDy)

True System

SINDy fitting





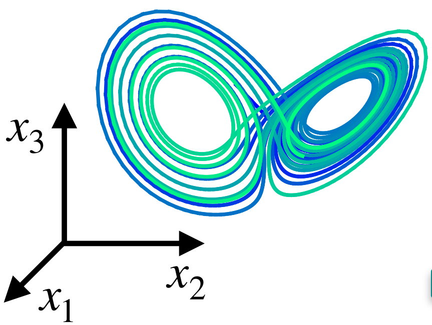
$$\dot{x}_1 = \sigma(x_2 - x_1)$$

$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$

$$\dot{x}_3 = x_1 x_2 - \beta x_3$$

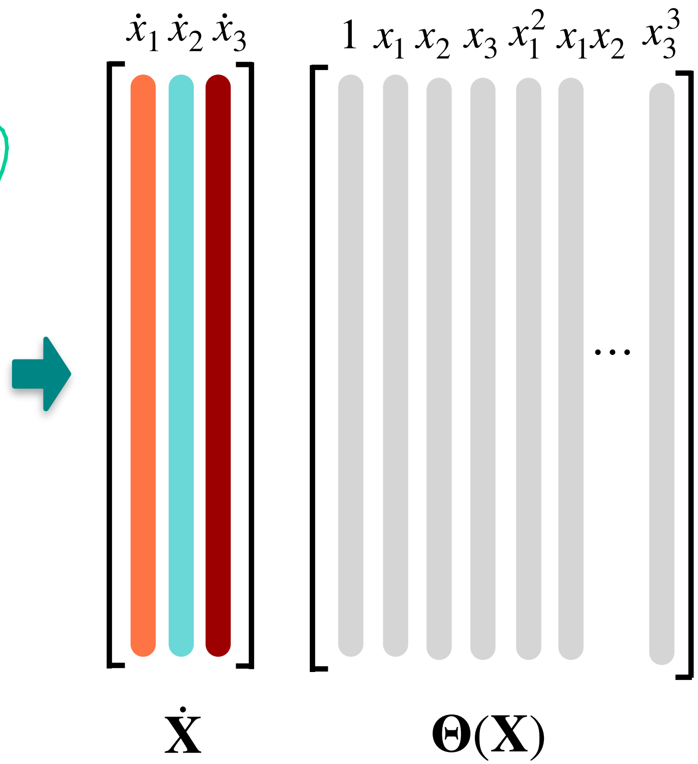
Sparse Identification of Nonlinear Dynamics (SINDy)

True System



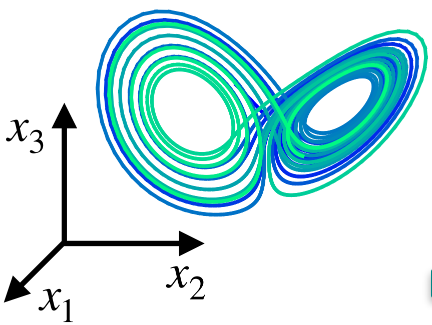
$$\dot{x}_1 = \sigma(x_2 - x_1)$$
$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$
$$\dot{x}_3 = x_1 x_2 - \beta x_3$$

SINDy fitting



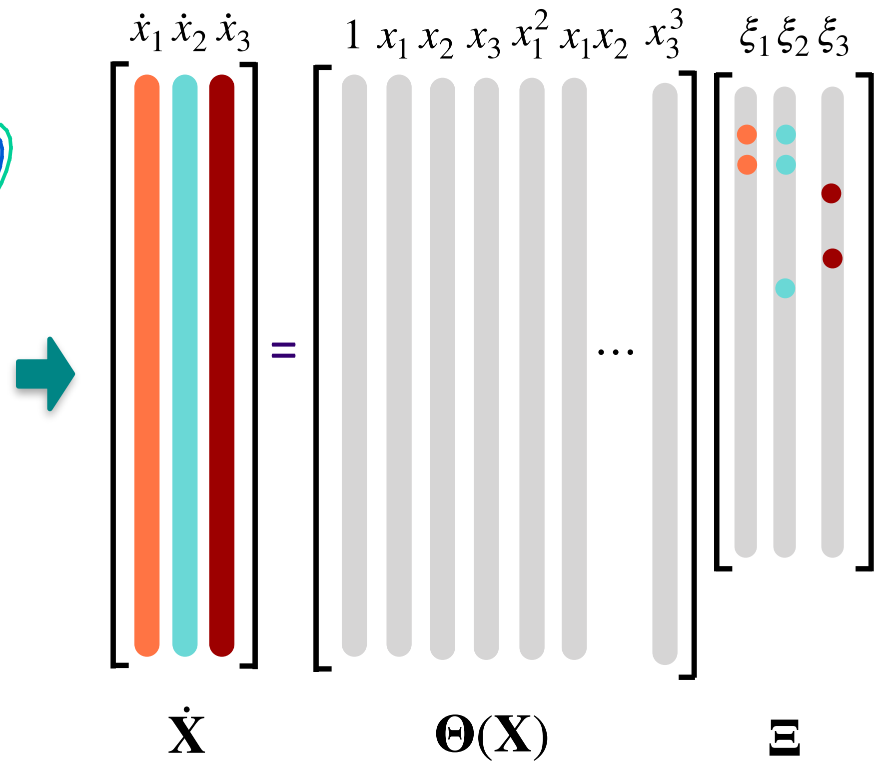
Sparse Identification of Nonlinear Dynamics (SINDy)

True System

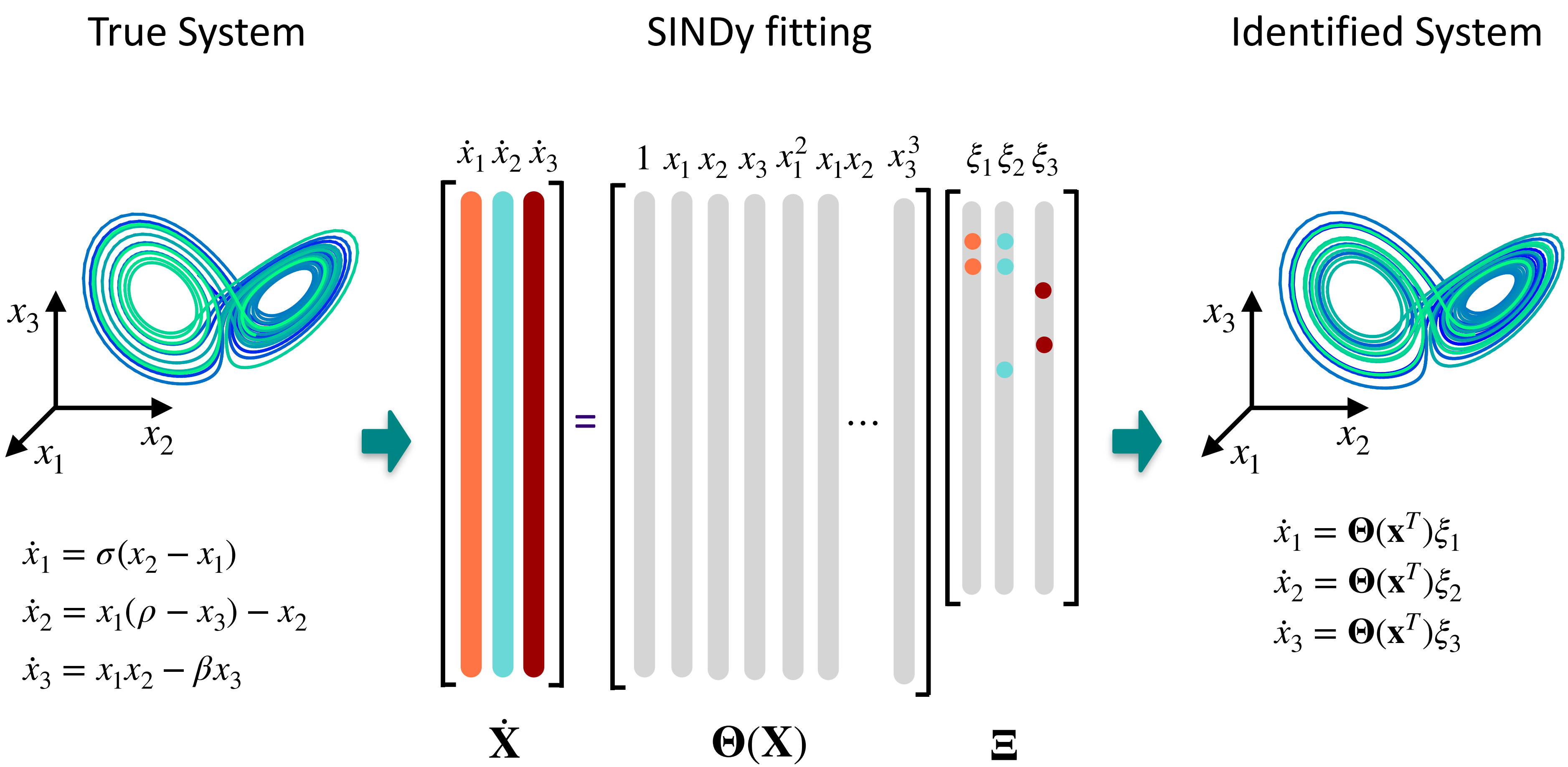


$$\dot{x}_1 = \sigma(x_2 - x_1)$$
$$\dot{x}_2 = x_1(\rho - x_3) - x_2$$
$$\dot{x}_3 = x_1 x_2 - \beta x_3$$

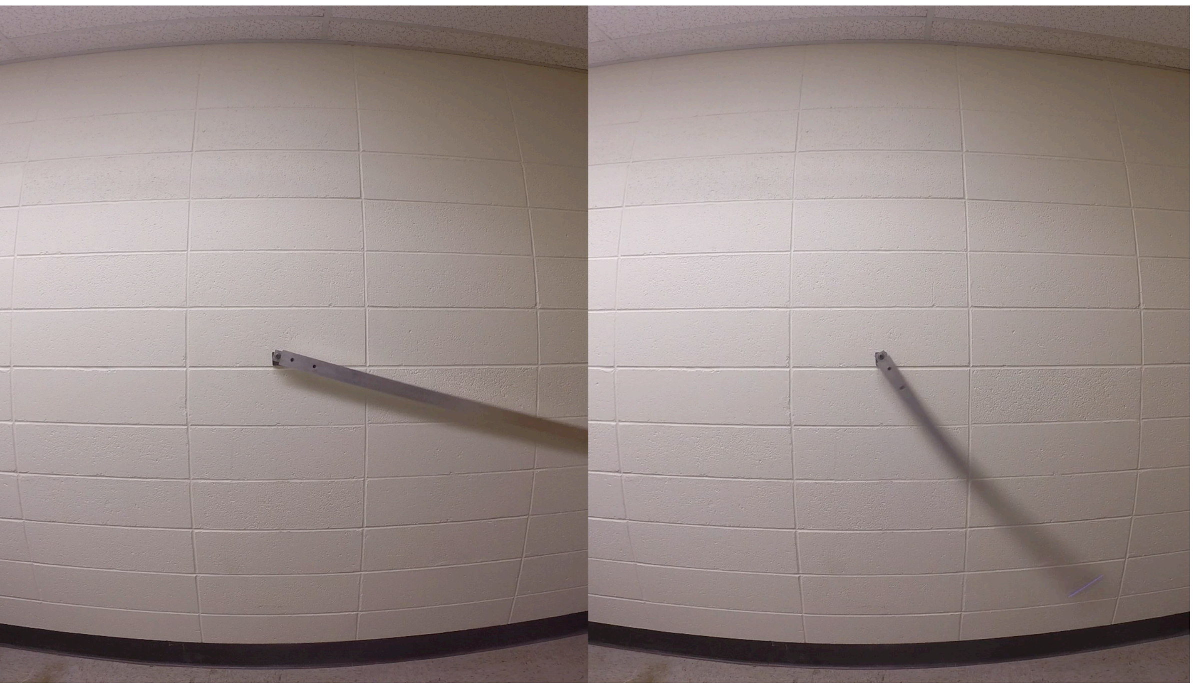
SINDy fitting

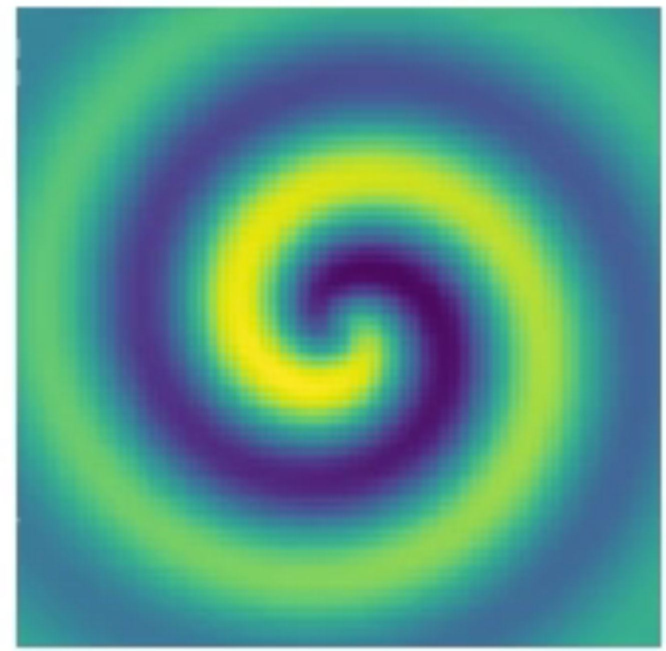


Sparse Identification of Nonlinear Dynamics (SINDy)

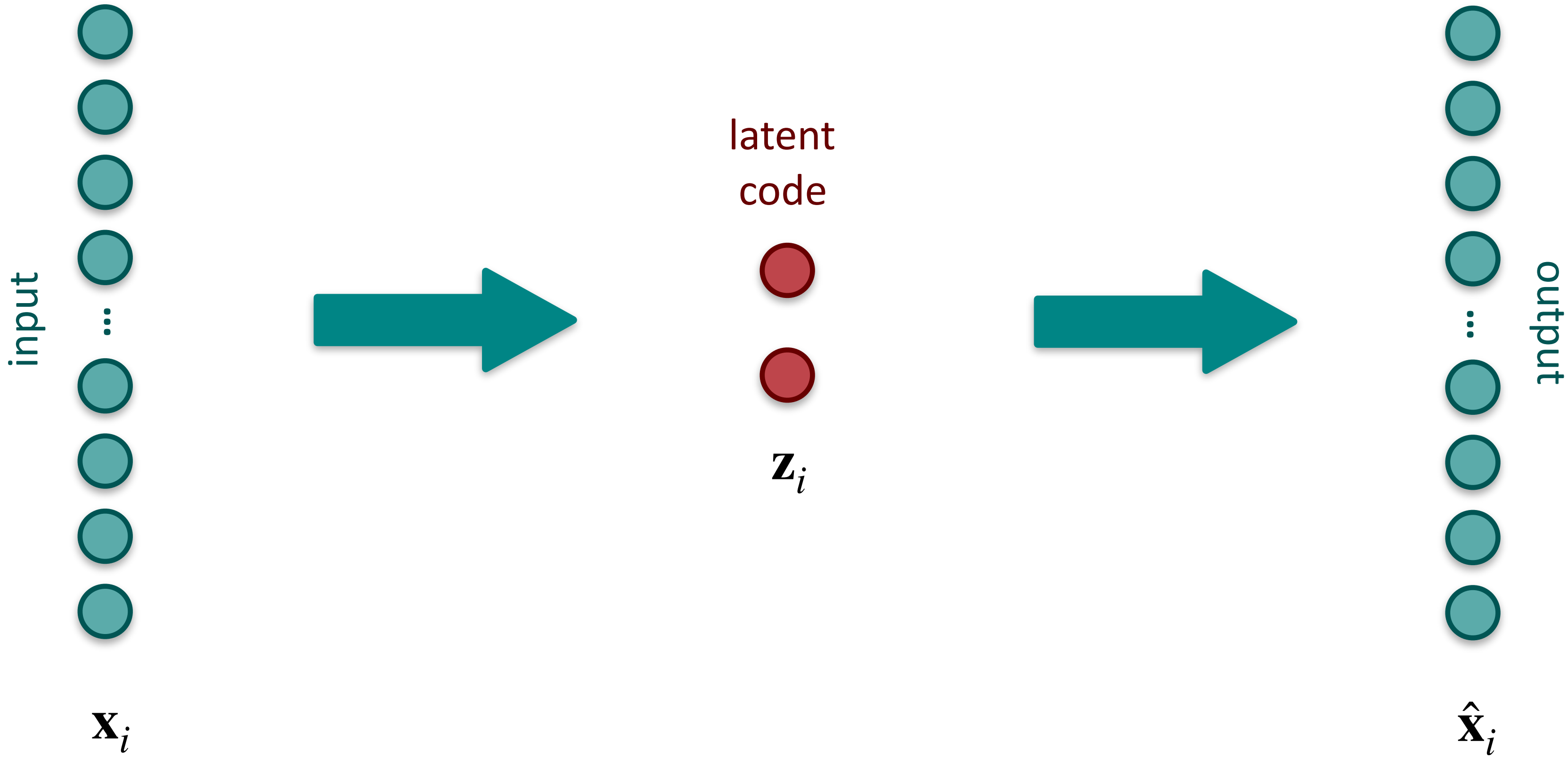


What if we don't know the right coordinates?



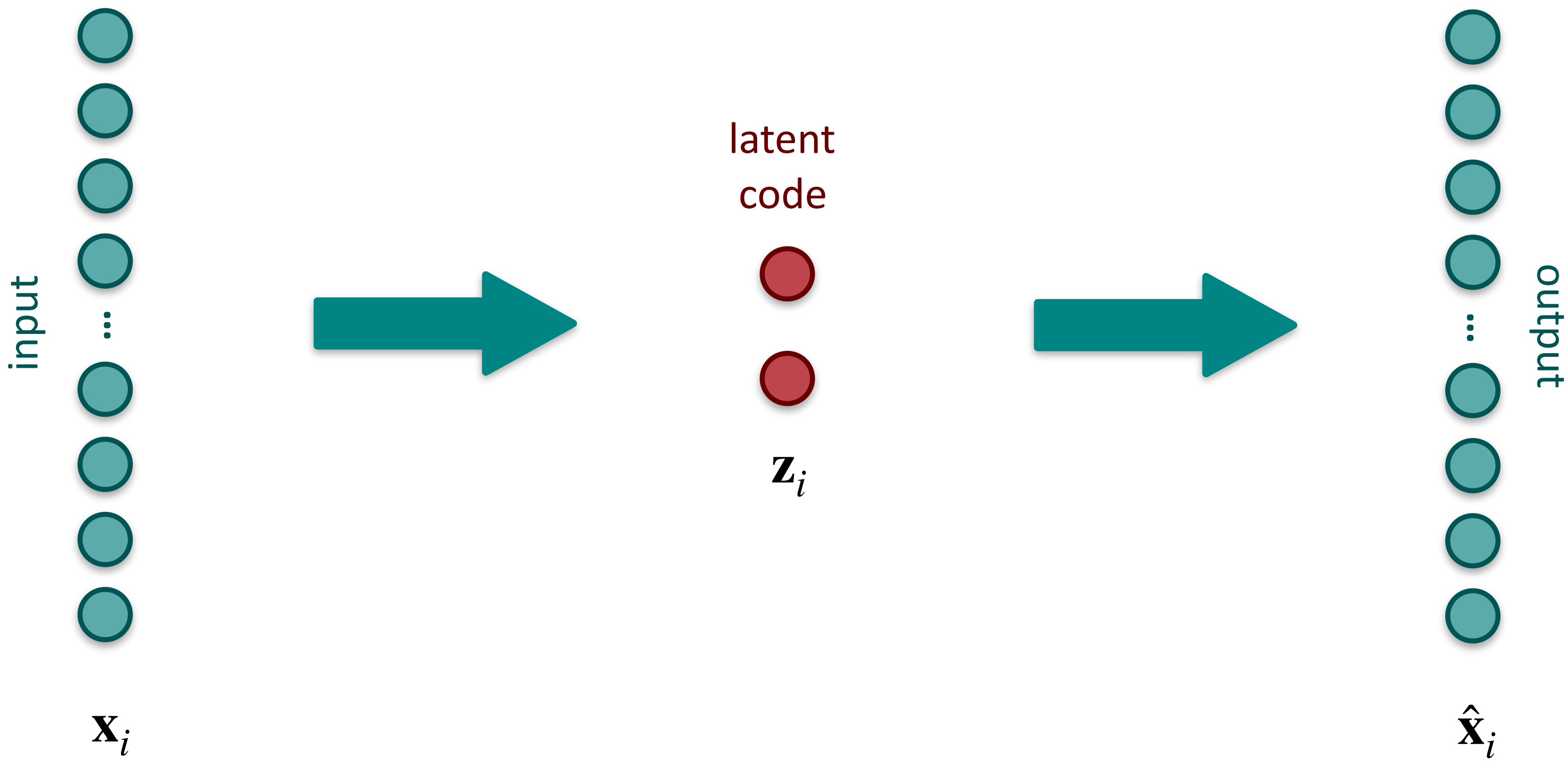


Autoencoder



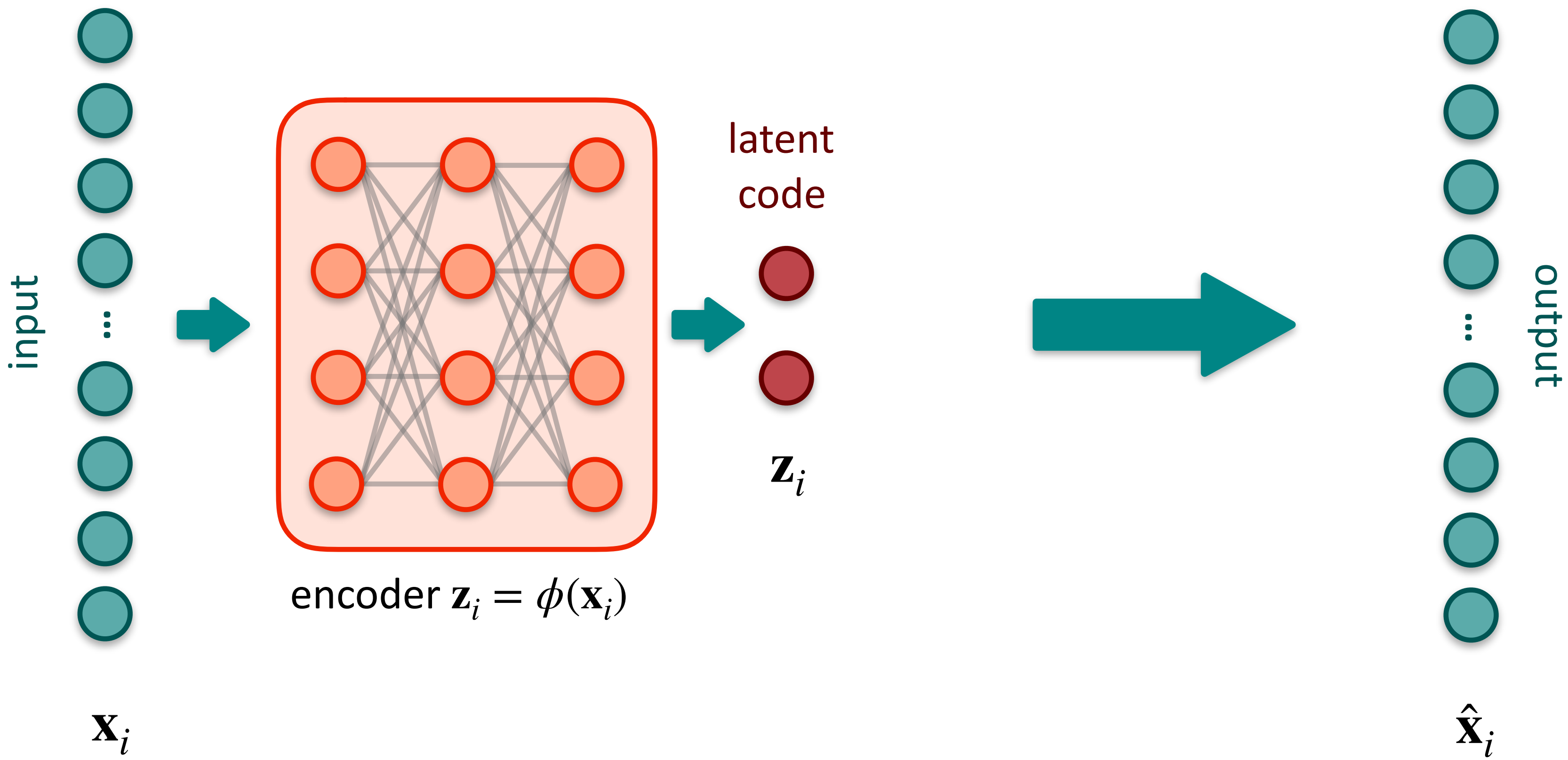
loss function: $$\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

Autoencoder



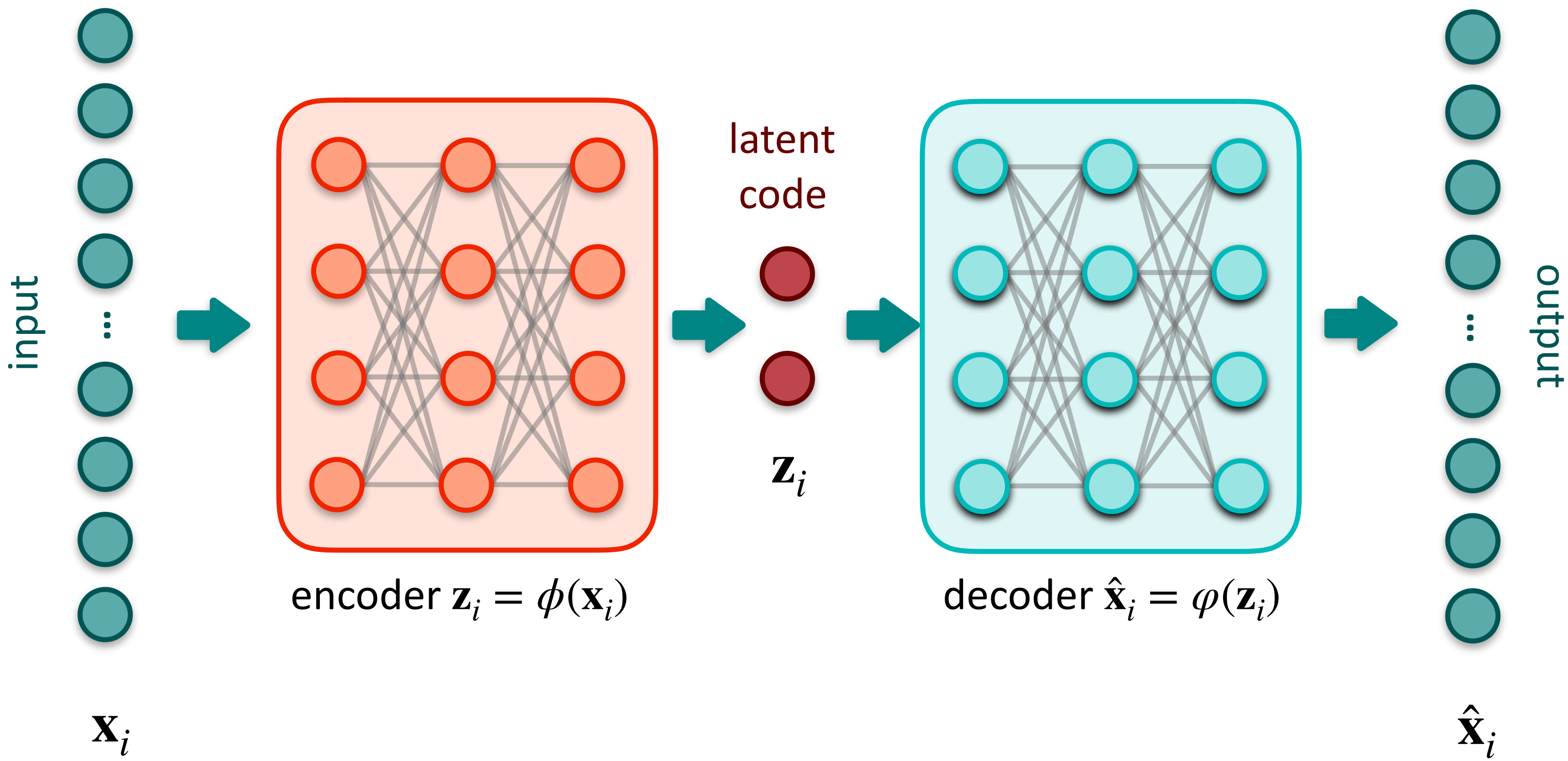
loss function: $$\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

Autoencoder



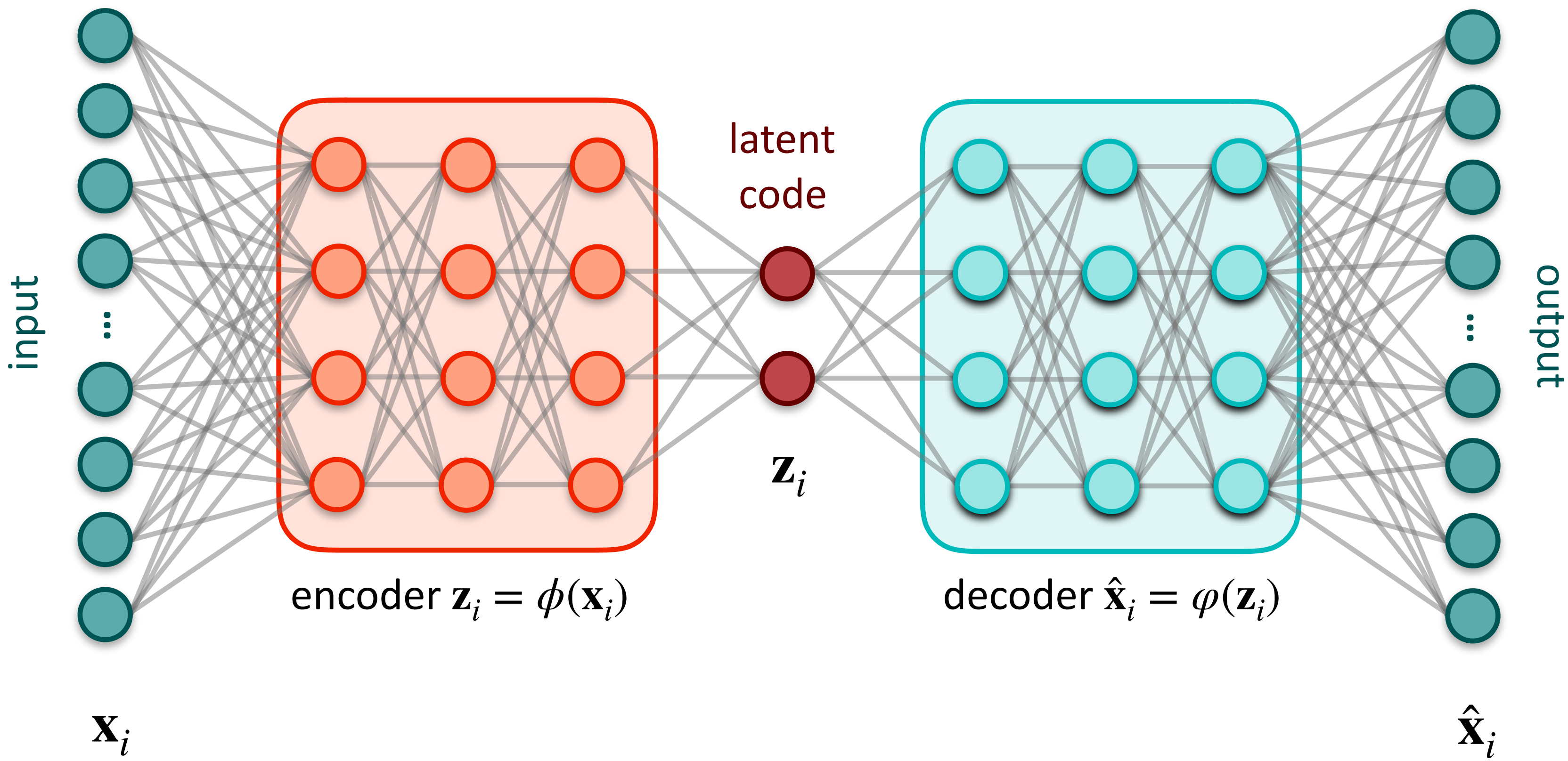
loss function: $\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$

Autoencoder



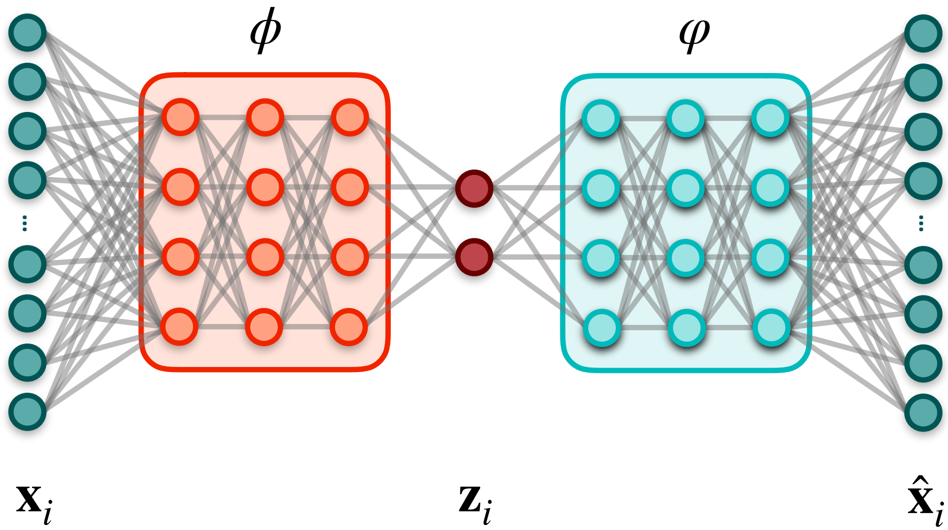
loss function: $$\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

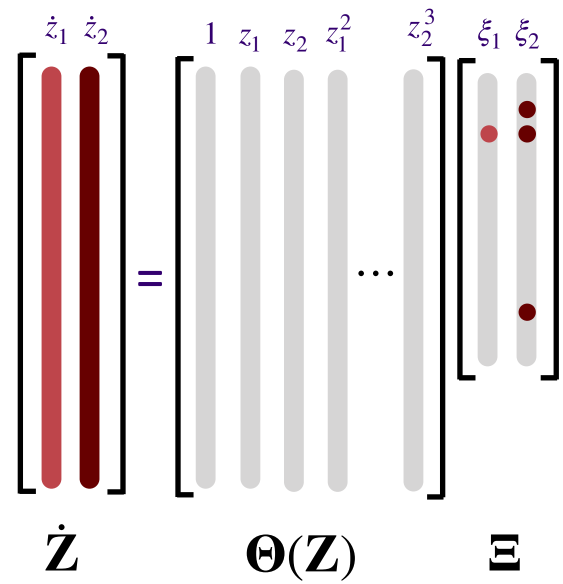
Autoencoder



loss function: $$\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

Autoencoder + SINDy

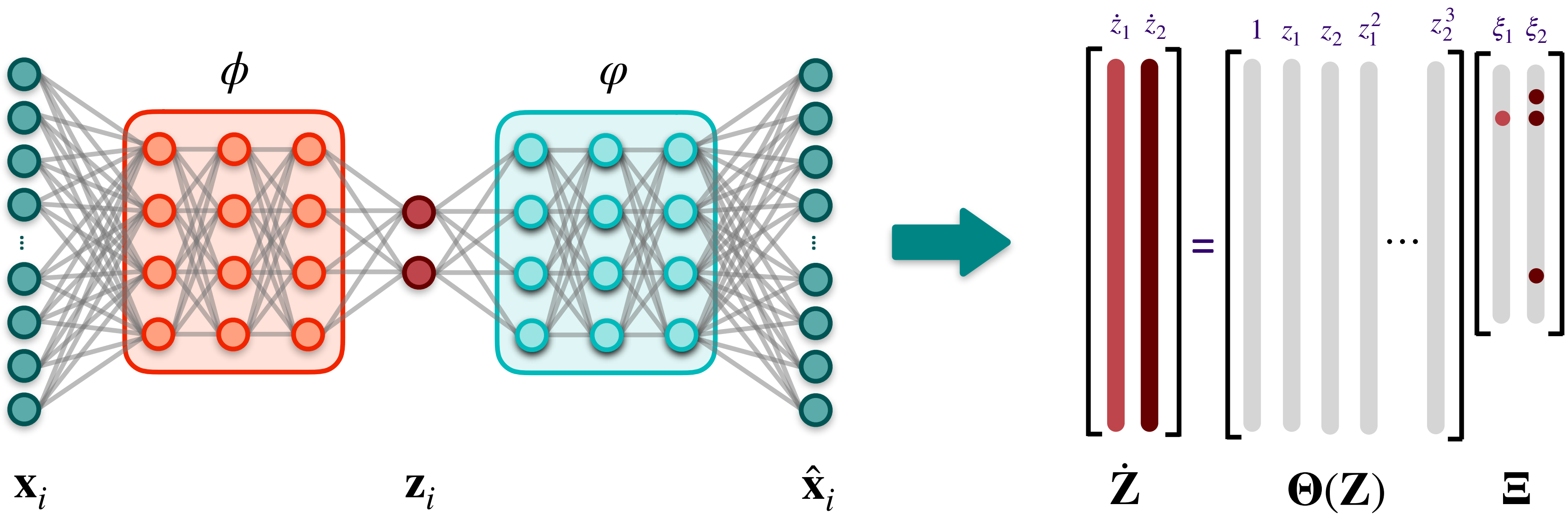




loss: $\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$

loss: $\frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{z}}_i - \mathbf{\Theta}(\mathbf{z}_i^T)\mathbf{\Xi}\|_2^2$

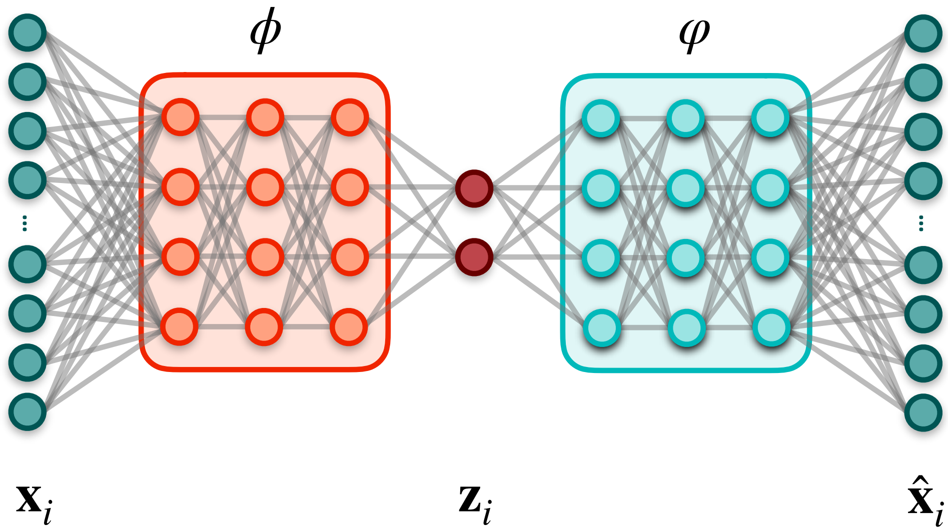
Autoencoder + SINDy

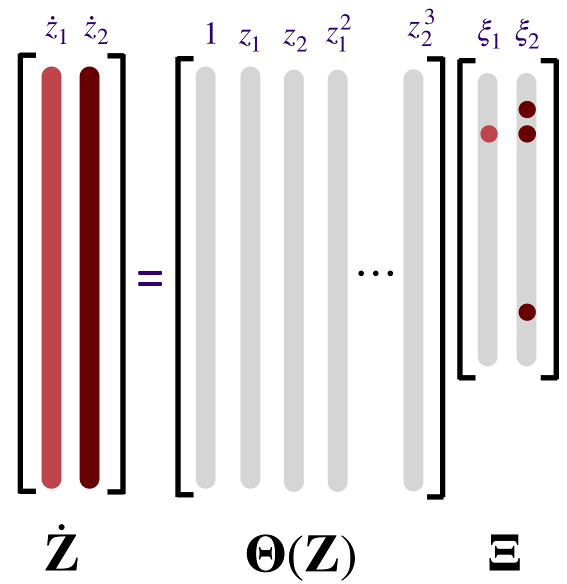


loss: $$\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

loss: $$\frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{z}}_i - \boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}\|_2^2$$

Autoencoder + SINDy

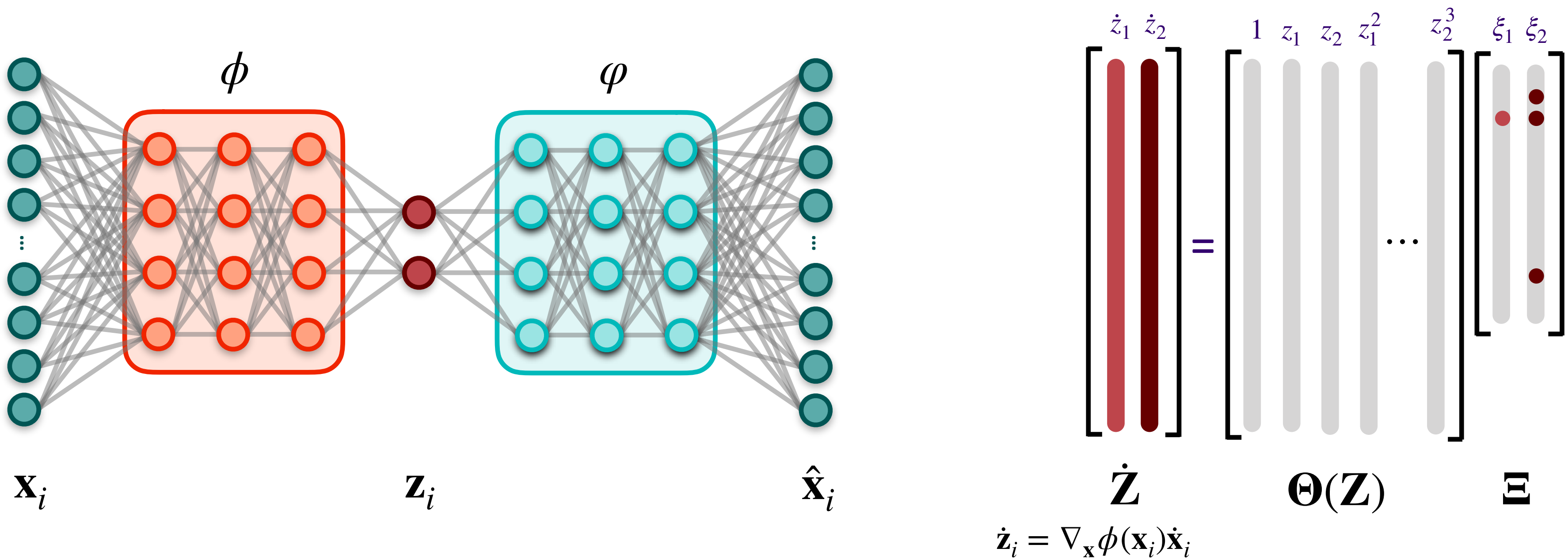




loss: $\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$

loss: $\frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{z}}_i - \mathbf{\Theta}(\mathbf{z}_i^T)\mathbf{\Xi}\|_2^2$

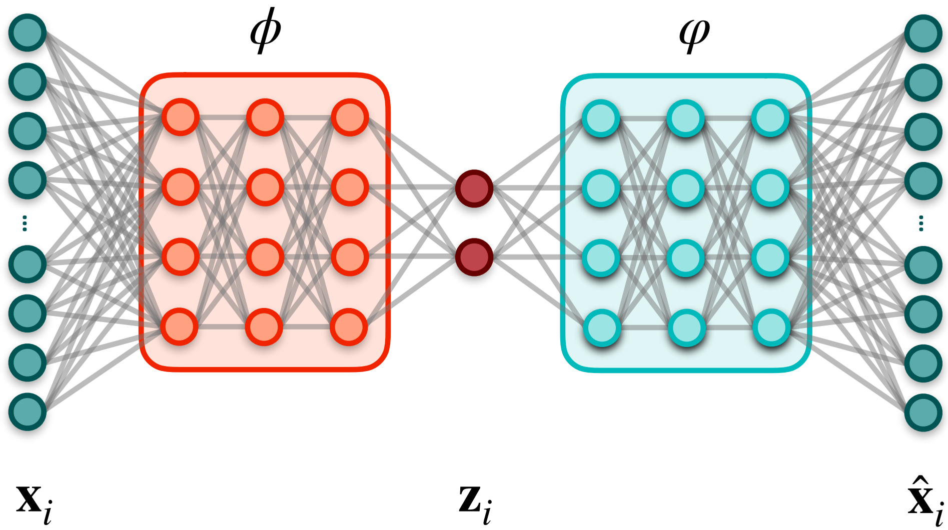
Autoencoder + SINDy

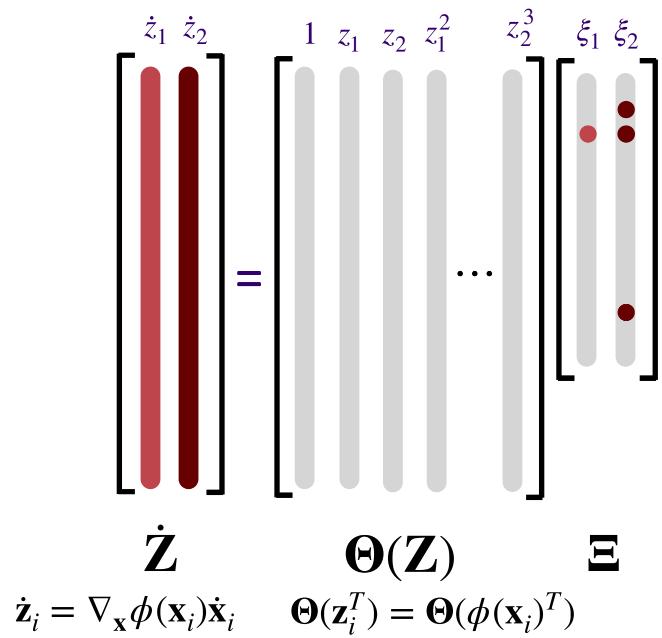


loss: $$\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

loss: $$\frac{1}{N}\sum_{i=1}^{N}\|\dot{\mathbf{z}}_i - \mathbf{\Theta}(\mathbf{z}_i^T)\mathbf{\Xi}\|_2^2$$

Autoencoder + SINDy

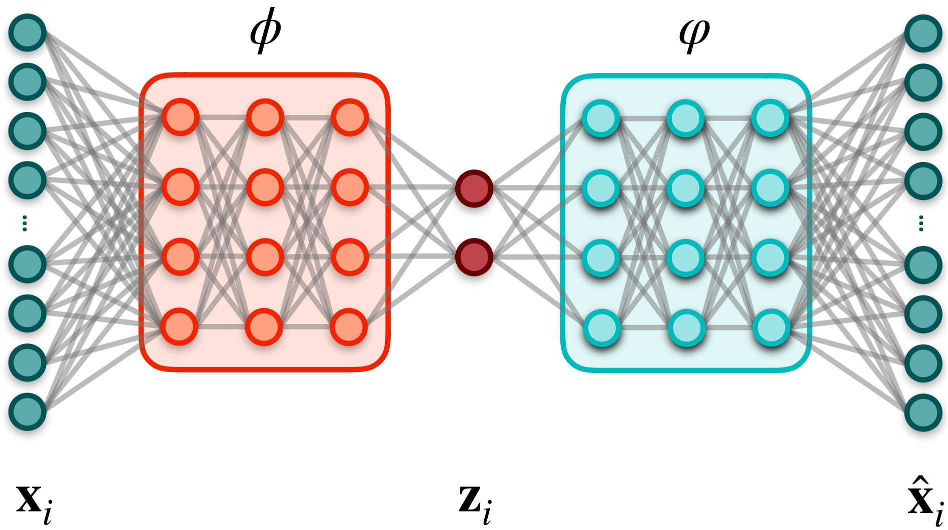


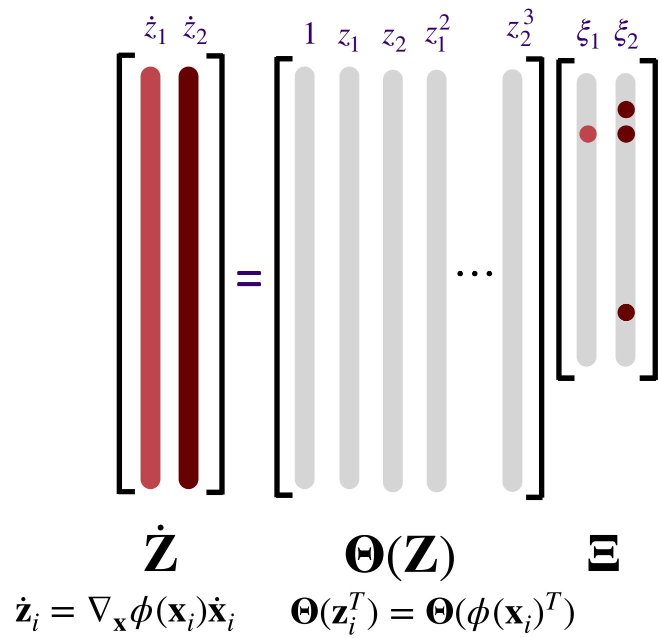


loss: $$\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2$$

loss: $$\frac{1}{N}\sum_{i=1}^{N}\|\dot{\mathbf{z}}_i - \mathbf{\Theta}(\mathbf{z}_i^T)\mathbf{\Xi}\|_2^2$$

Autoencoder + SINDy





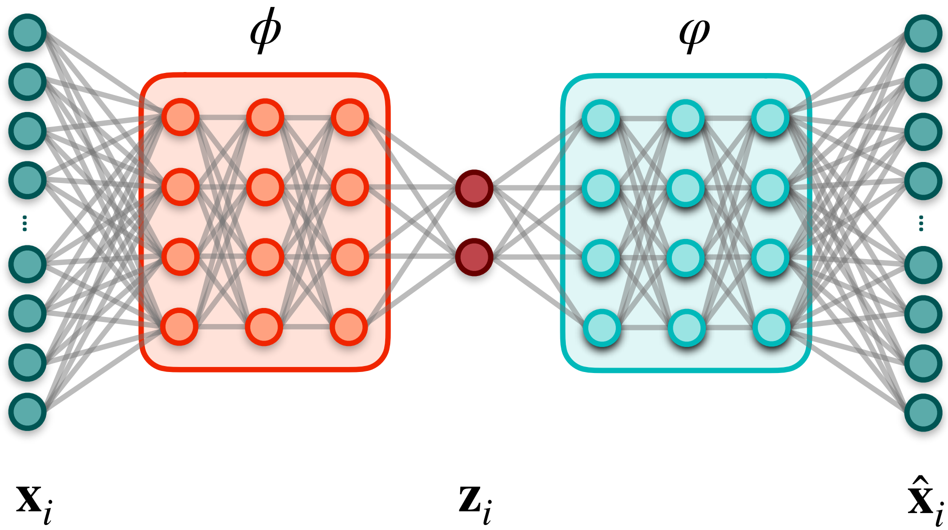
loss:

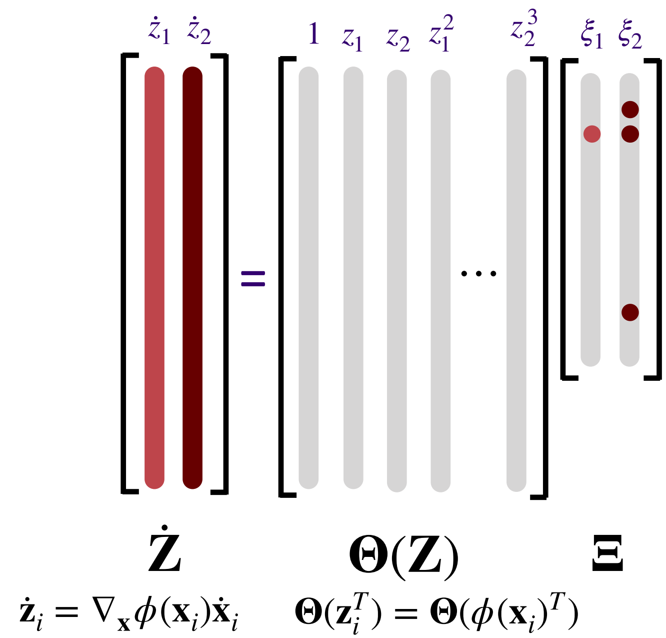
$$\lambda_1 \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \varphi(\phi(\mathbf{x}_i))\|_2^2 \quad + \quad \lambda_2 \frac{1}{N}\sum_{i=1}^{N} \|\nabla_{\mathbf{x}}\phi(\mathbf{x}_i)\dot{\mathbf{x}}_i - \boldsymbol{\Theta}(\phi(\mathbf{x}_i)^T)\boldsymbol{\Xi}\|_2^2$$

autoencoder component

SINDy component

Autoencoder + SINDy



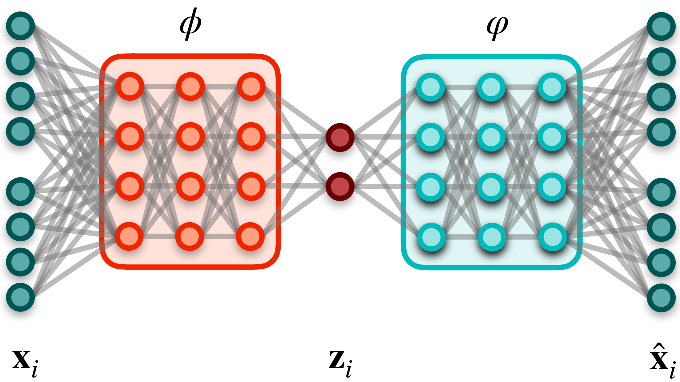


loss: $$\lambda_1 \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \varphi(\phi(\mathbf{x}_i))\|_2^2 \; + \; \lambda_2 \frac{1}{N}\sum_{i=1}^{N} \|\nabla_{\mathbf{x}}\phi(\mathbf{x}_i)\dot{\mathbf{x}}_i - \boldsymbol{\Theta}(\phi(\mathbf{x}_i)^T)\boldsymbol{\Xi}\|_2^2 + \lambda_3\|\boldsymbol{\Xi}\|_1$$

autoencoder component

SINDy component

Autoencoder + SINDy



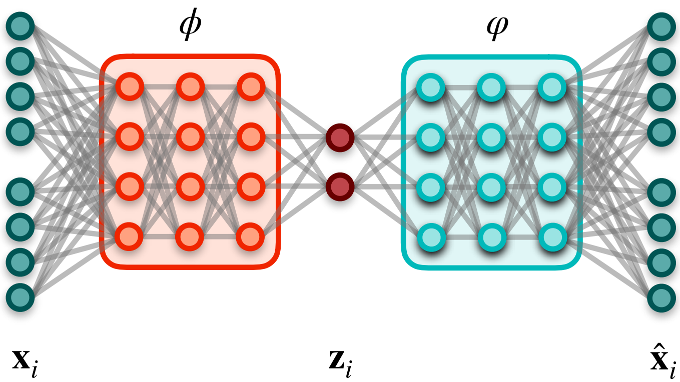
loss:

$$\lambda_1 \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 + \lambda_2 \frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{z}}_i - \boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}\|_2^2 + \lambda_3 \|\Xi\|_1$$

L_1 L_2 L_3

> ***Issue:*** training shrinks norm of **z** to minimize loss function

Autoencoder + SINDy



loss:

$$\lambda_1 \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 + \lambda_2 \frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{z}}_i - \boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}\|_2^2 + \lambda_3 \|\boldsymbol{\Xi}\|_1$$

L_1 L_2 L_3

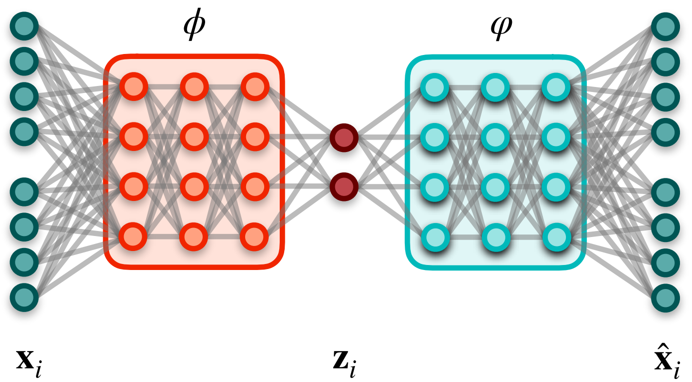
> ***Issue:*** training shrinks norm of **z** to minimize loss function

> ***Solution:*** use the following to enforce SINDy loss

new L_2 :

$$\frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{x}}_i - \nabla_{\mathbf{z}}\varphi(\mathbf{z}_i)\underbrace{\boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}}_{\dot{\mathbf{z}}_i}\|_2^2 = \frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{x}}_i - \nabla_{\mathbf{z}}\varphi(\phi(\mathbf{x}_i))\underbrace{\boldsymbol{\Theta}(\phi(\mathbf{x}_i)^T)\boldsymbol{\Xi}}_{\dot{\mathbf{z}}_i}\|_2^2$$

Autoencoder + SINDy



loss:

$$\lambda_1 \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 + \lambda_2 \frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{z}}_i - \boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}\|_2^2 + \lambda_3 \|\boldsymbol{\Xi}\|_1$$

L_1 L_2 L_3

> ***Issue:*** training shrinks norm of **z** to minimize loss function

> ***Solution:*** use the following to enforce SINDy loss

new L_2 :

$$\frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{x}}_i - \nabla_{\mathbf{z}}\varphi(\mathbf{z}_i)\underbrace{\boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}}_{\dot{\mathbf{z}}_i}\|_2^2 = \frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{x}}_i - \nabla_{\mathbf{z}}\varphi(\phi(\mathbf{x}_i))\underbrace{\boldsymbol{\Theta}(\phi(\mathbf{x}_i)^T)\boldsymbol{\Xi}}_{\dot{\mathbf{z}}_i}\|_2^2$$

> New loss function:

$$\lambda_1 \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 + \lambda_2 \frac{1}{N}\sum_{i=1}^{N} \|\dot{\mathbf{x}}_i - \nabla_{\mathbf{z}}\varphi(\mathbf{z}_i)\boldsymbol{\Theta}(\mathbf{z}_i^T)\boldsymbol{\Xi}\|_2^2 + \lambda_3 \|\boldsymbol{\Xi}\|_1$$

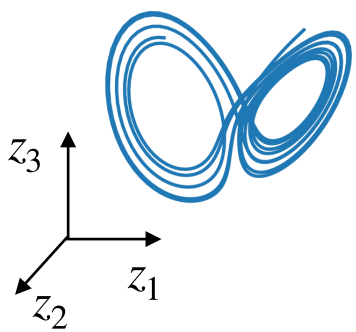
Achieving sparsity

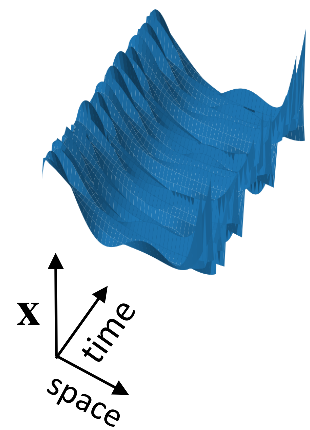
> With L1 penalty alone, get model that has many very small coefficients but is not truly sparse

$$\lambda_1 \frac{1}{N} \sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2 + \lambda_2 \frac{1}{N} \sum_{i=1}^{N} \|\dot{\mathbf{x}}_i - \nabla_{\mathbf{z}} \varphi(\mathbf{z}_i) \boldsymbol{\Theta}(\mathbf{z}_i^T) \boldsymbol{\Xi}\|_2^2 + \lambda_3 \|\boldsymbol{\Xi}\|_1$$

> Instead combine L1 penalty with sequential thresholding

Example problem





$$\mathbf{x}(t) = \mathbf{A}\mathbf{z}(t) + \mathbf{B}\begin{pmatrix} z_1^3(t) \\ z_2^3(t) \\ z_3^3(t) \end{pmatrix}$$

$$\mathbf{x}(t) \in \mathbb{R}^{128}$$

$$\mathbf{A}, \mathbf{B} \in \mathbb{R}^{128 \times 3}$$

Example problem

Lorenz model

Equations

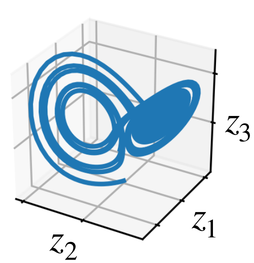
$$\dot{z}_1 = -10z_1 + 10z_2$$

$$\dot{z}_2 = 28z_1 - z_2 - z_1z_3$$

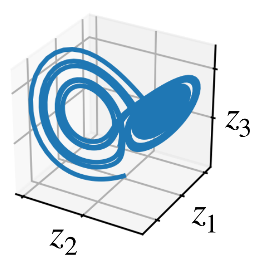
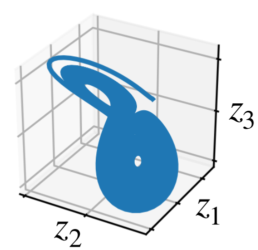
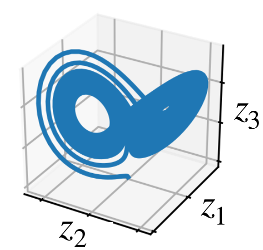
$$\dot{z}_3 = -2.7z_3 + z_1z_2$$

Coefficient Matrix Ξ

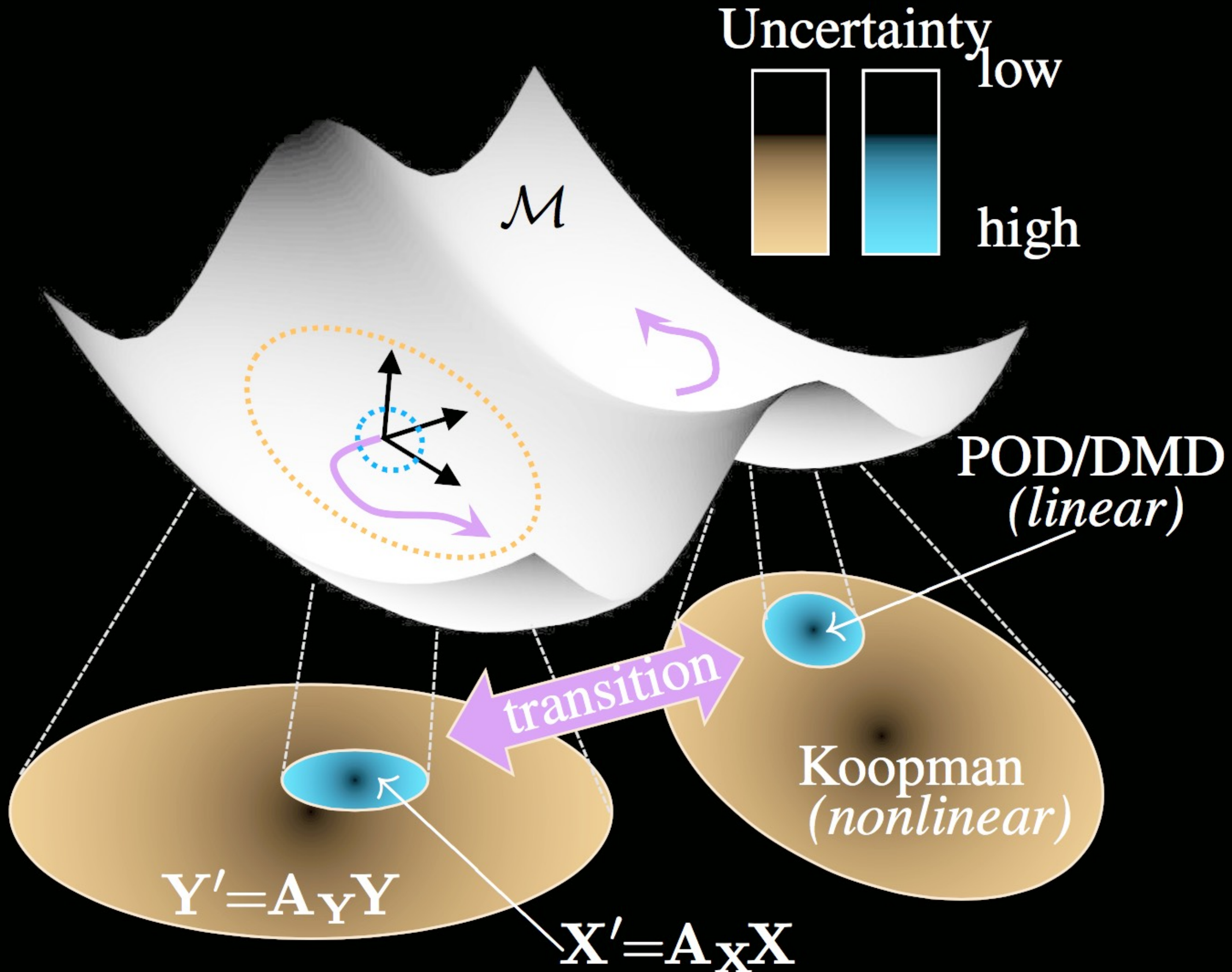
Dynamics



Example problem

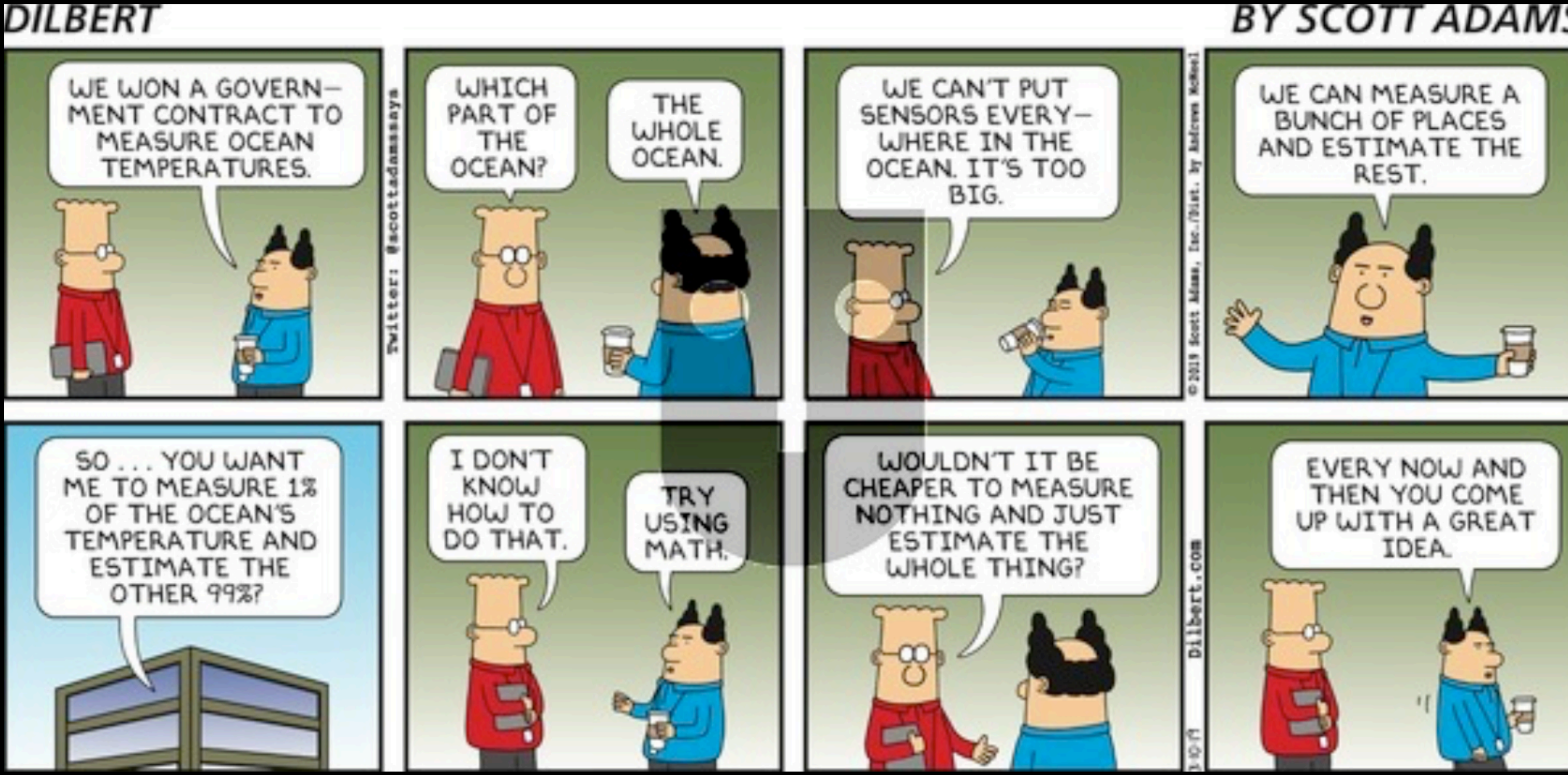
	Lorenz model	Discovered model	Discovered model (transformed)
Equations	$\dot{z}_1 = -10z_1 + 10z_2$ $\dot{z}_2 = 28z_1 - z_2 - z_1z_3$ $\dot{z}_3 = -2.7z_3 + z_1z_2$	$\dot{z}_1 = -8.5z_2z_3$ $\dot{z}_2 = 9.2 - 2.9z_2 + 1.1z_1z_3$ $\dot{z}_3 = -8.8z_1 - 10.3z_3$	$\dot{z}_1 = -10.2z_1 + 8.8z_2$ $\dot{z}_2 = 26.7z_1 - 8.5z_1z_3$ $\dot{z}_3 = -2.9z_3 + 1.1z_2$
Coefficient Matrix Ξ			
Dynamics			

Koopman and UQ



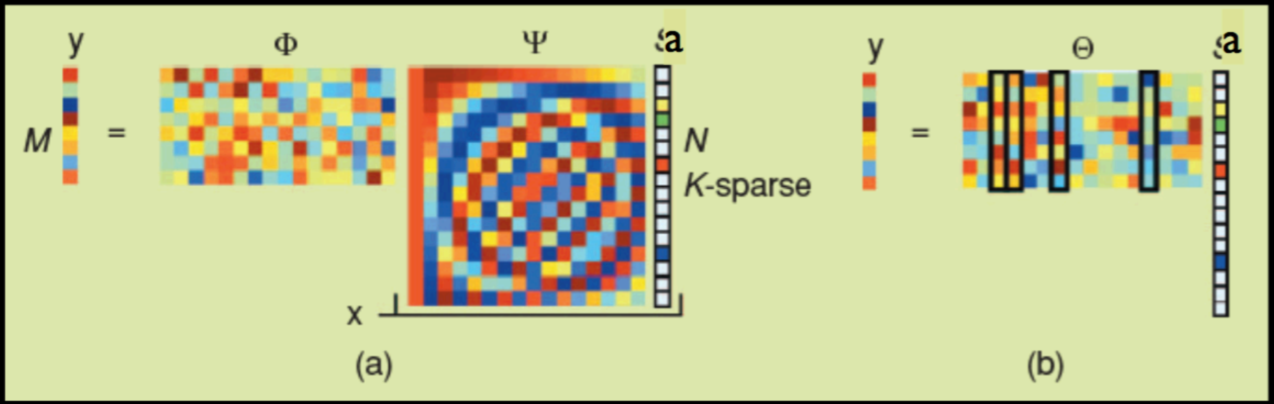
Measurement and Sensors

Randomized Linear Algebra
&
Promoting Sparsity



Dilbert (Scott Adams) — March 10, 2019

Compressive Sensing: A Cartoon



from Baraniuk, 2007.

IN: low-dimensional measurements

random measurement matrix

OUT: sparse coefficients

$$y = \Phi x = \Phi\Psi a = \Theta a$$

transform basis

IMPORTANT: measurement matrix must be incoherent with respect to the transform basis

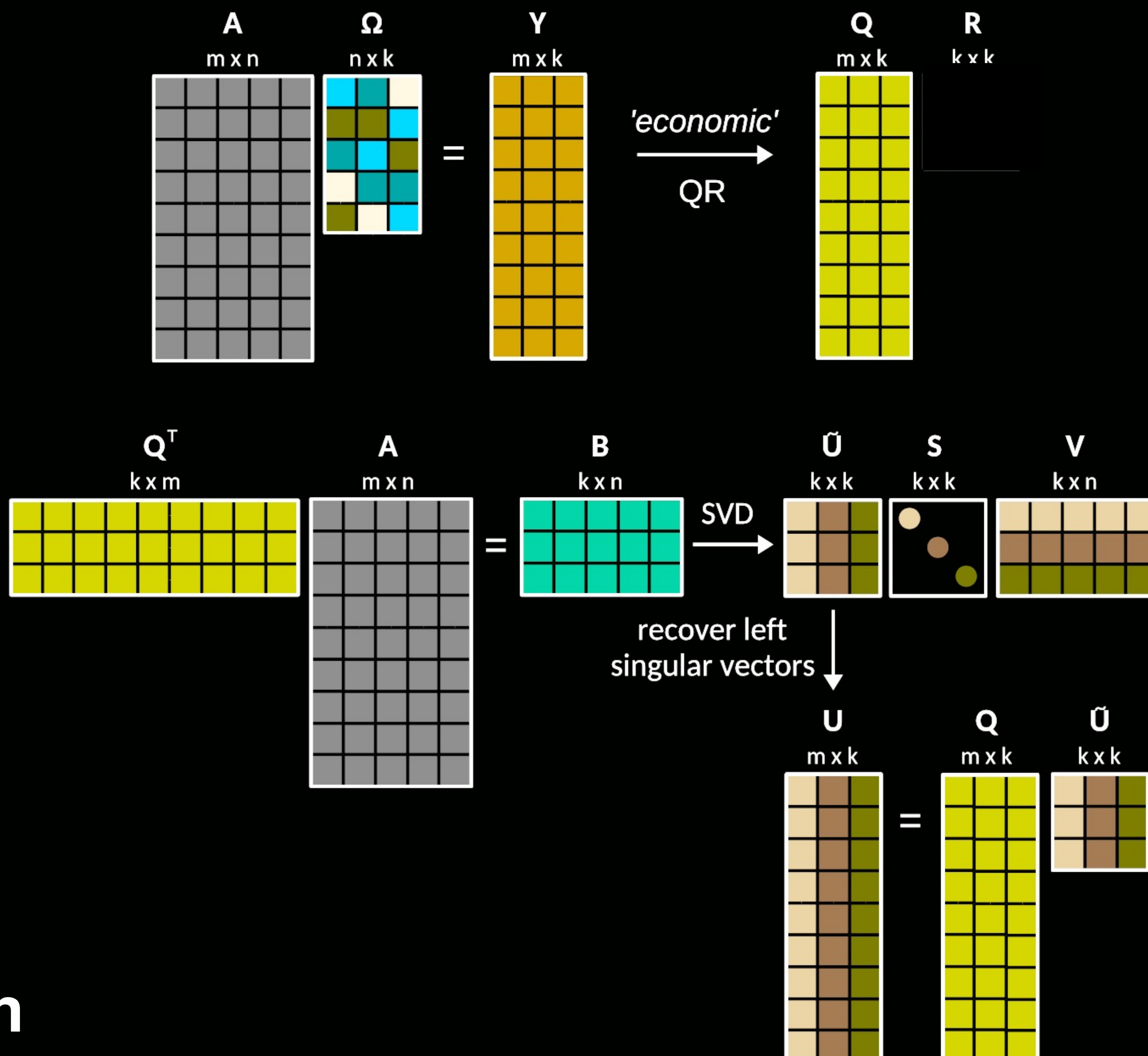
To reconstruct:

$$\text{minimize } ||a||_1,$$
$$\text{such that } y = \Theta a$$

Proofs by:

- Candès, Romberg & Tao, 2006.
- Donoho, 2006.

Randomized Linear Algebra



Ben Erichson

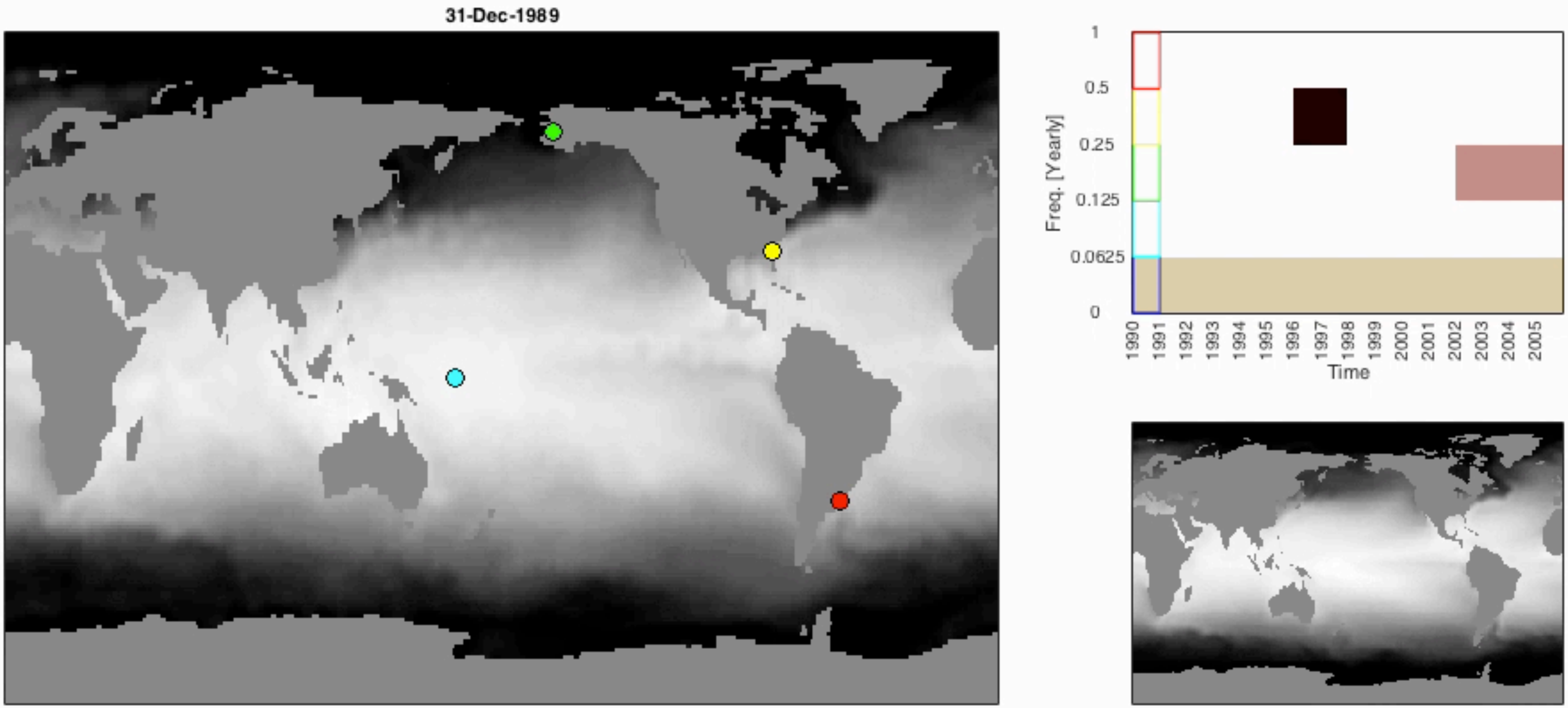
Gappy Methods

```
% QR sensor selection, p=K
[Q,R,pivot] = qr(Psi_r','vector');
pivot = pivot(1:K)
```

Everson & Sirovich (1995)
Willcox (2005)
Karniadakis co-workers (2009)
Maday, Patera et al & Sorenson et al (2010, 2012)
Gugerkin & Drmac (2015)
Manohar, Brunton, Kutz & Brunton (2017)

Krithika Manohar

Respect Multiscale Features



Mathematical Framework

State space $\boldsymbol{x} \in \mathbb{R}^m$

Measurements $\boldsymbol{s} \in \mathbb{R}^p$

$p \ll m$

Mapping

$$\widehat{\boldsymbol{x}} = \mathcal{F}(\boldsymbol{s})$$

Approximate the full state space from limited measurements

Optimization

$$\mathcal{F} \in \underset{\widetilde{\mathcal{F}} \in \mathscr{F}}{\arg\min} \sum_{i=1}^{n} \left\| \boldsymbol{x}_i - \widetilde{\mathcal{F}}(\boldsymbol{s}_i) \right\|_2^2$$

training set $\{\boldsymbol{x}_i, \boldsymbol{s}_i\}_{i=1}^{n}$ with n examples

Linear Maps

Singular value decomposition

$$\boldsymbol{X} \overset{\text{rank-}k}{\approx} \boldsymbol{\Phi}\,\boldsymbol{\Sigma}\,\boldsymbol{V}^*$$

Data

$$\boldsymbol{X} = (\boldsymbol{x}_1 \ldots \boldsymbol{x}_n)$$

Linear measurements H

$$\boldsymbol{s} = \boldsymbol{H}\,\boldsymbol{x} \approx \boldsymbol{H}\,\boldsymbol{\Phi}\,\boldsymbol{\nu}$$

Optimize (least-squares)

$$\boldsymbol{\nu} \in \arg\min_{\widetilde{\boldsymbol{\nu}}} \|\boldsymbol{s} - \boldsymbol{H}\,\boldsymbol{\Phi}\,\widetilde{\boldsymbol{\nu}}\|_2^2$$

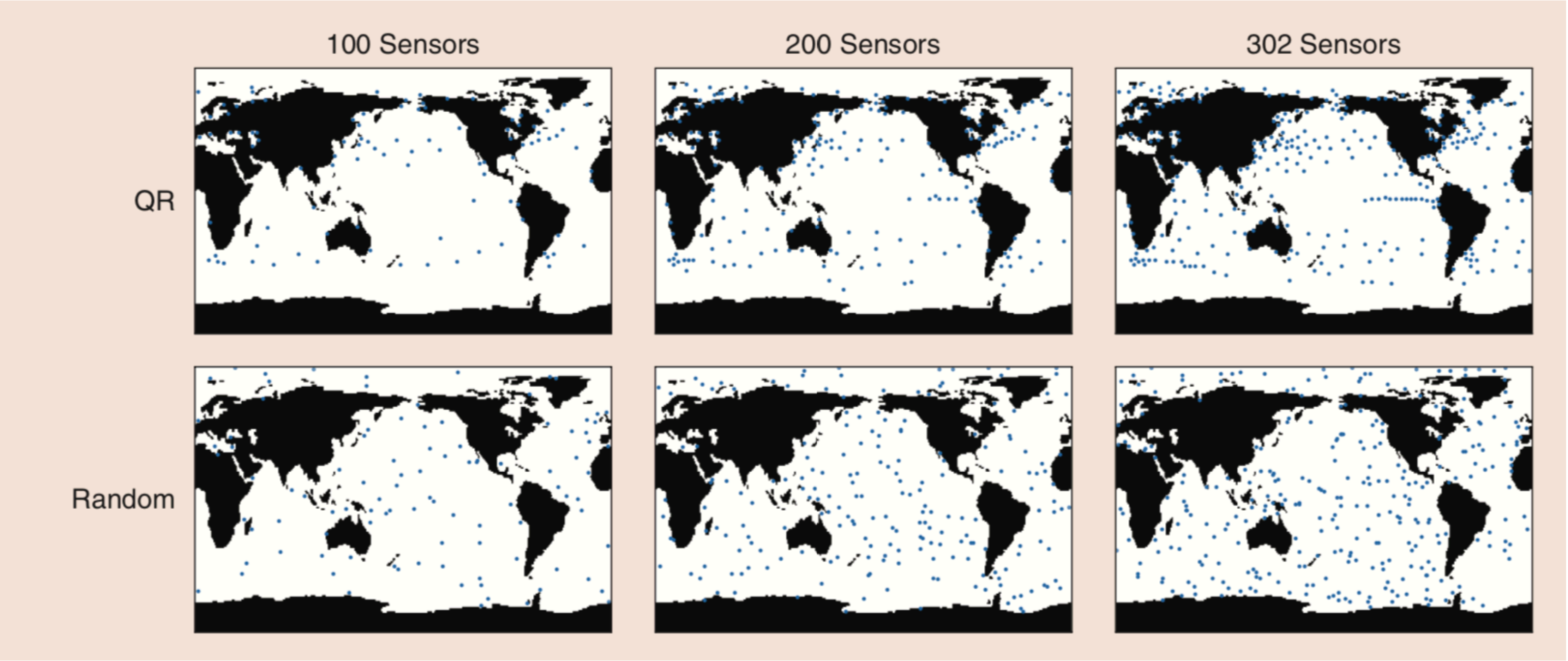
$$\boldsymbol{\nu} = (\boldsymbol{H}\,\boldsymbol{\Phi})^+\ \boldsymbol{s}$$

$$\boldsymbol{x} \approx \widehat{\boldsymbol{x}} = \boldsymbol{\Phi}\,\boldsymbol{\nu}$$

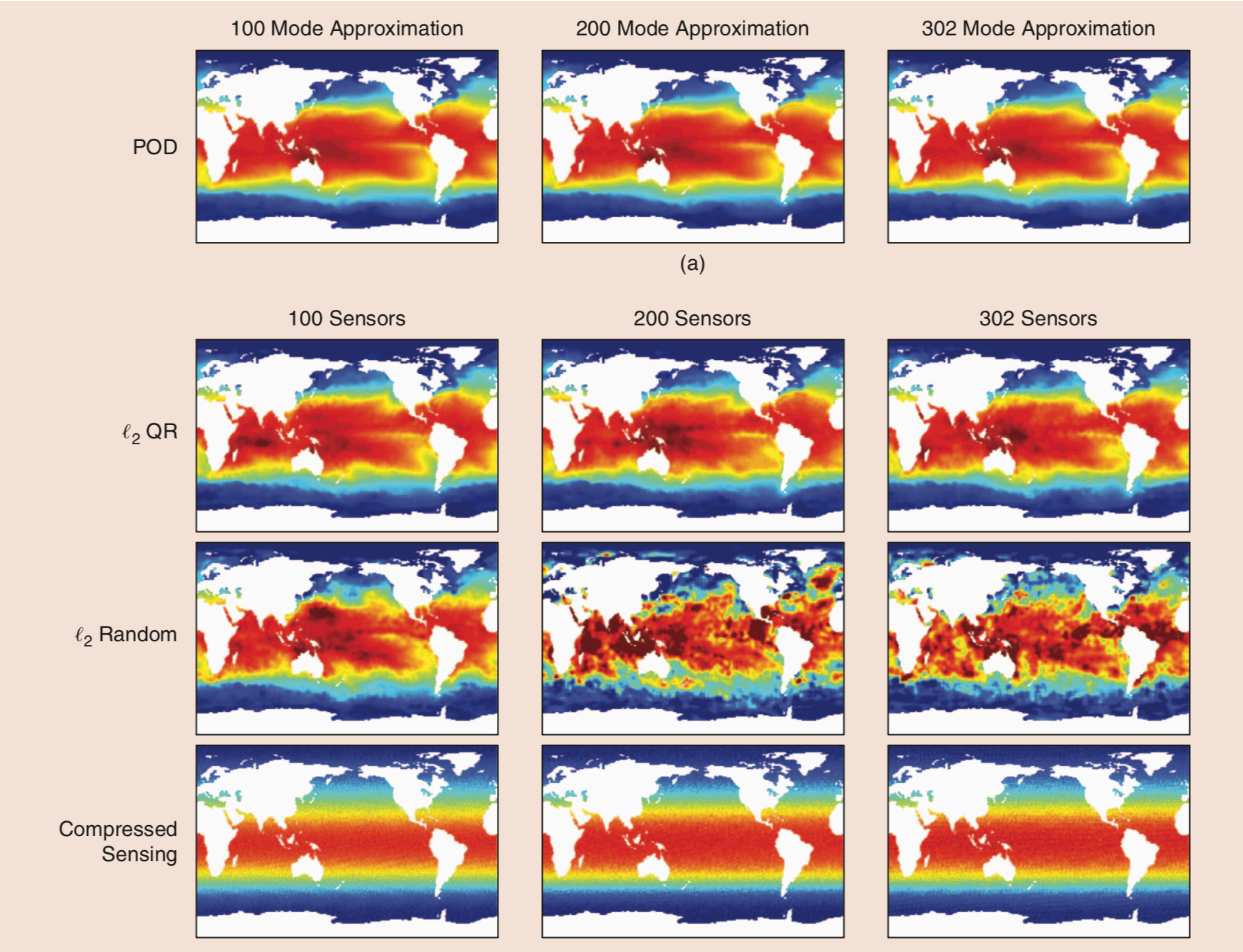
Optimal Placement

Point measurements $$H = [\mathbf{e}_{\gamma_1} \ \mathbf{e}_{\gamma_2} \ \ldots \ \mathbf{e}_{\gamma_p}]^T$$

Optimal Sensors via QR pivots



Manohar, Kutz & Brunton (2018) IEEE Control Systems Magazine


100 Mode Approximation
200 Mode Approximation
302 Mode Approximation
POD
(a)
100 Sensors
200 Sensors
302 Sensors
ℓ_2 QR
ℓ_2 Random
Compressed Sensing

Optimal Placement with Cost

Modify Optimization

$$J = \{j_1, \ldots, j_l\}$$ Measurement indices

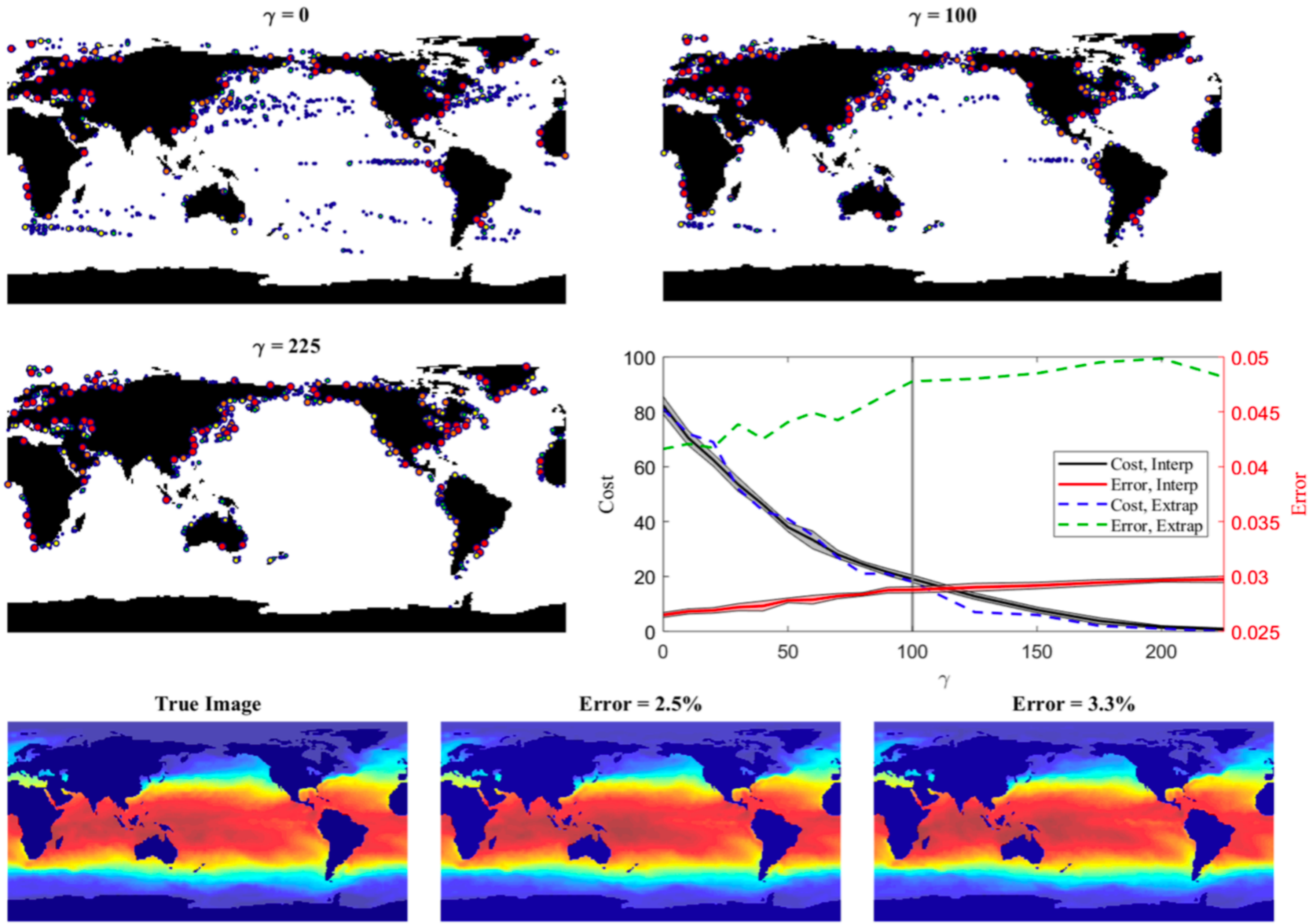
Cost Function

$$\hat{J} = \arg\min_{J} e(J) \text{ s.t. } \sum_{j \in J} \eta_j \leq b \text{ and } \|\hat{\mathbf{T}}(J)\|_{\infty,\text{vec}} \leq s$$

Fitting of Data

$$\hat{\mathbf{T}}(J) = \arg\min_{\mathbf{T} \in \mathbb{R}^{l \times n}} \|\mathbf{X} - \mathbf{X}_{\cdot J}\mathbf{T}\|_F$$

Clark, Askham, Brunton & Kutz (2019) IEEE Sensors



γ = 0
γ = 100
γ = 225
Cost
Error
γ
Cost, Interp
Error, Interp
Cost, Extrap
Error, Extrap
True Image
Error = 2.5%
Error = 3.3%

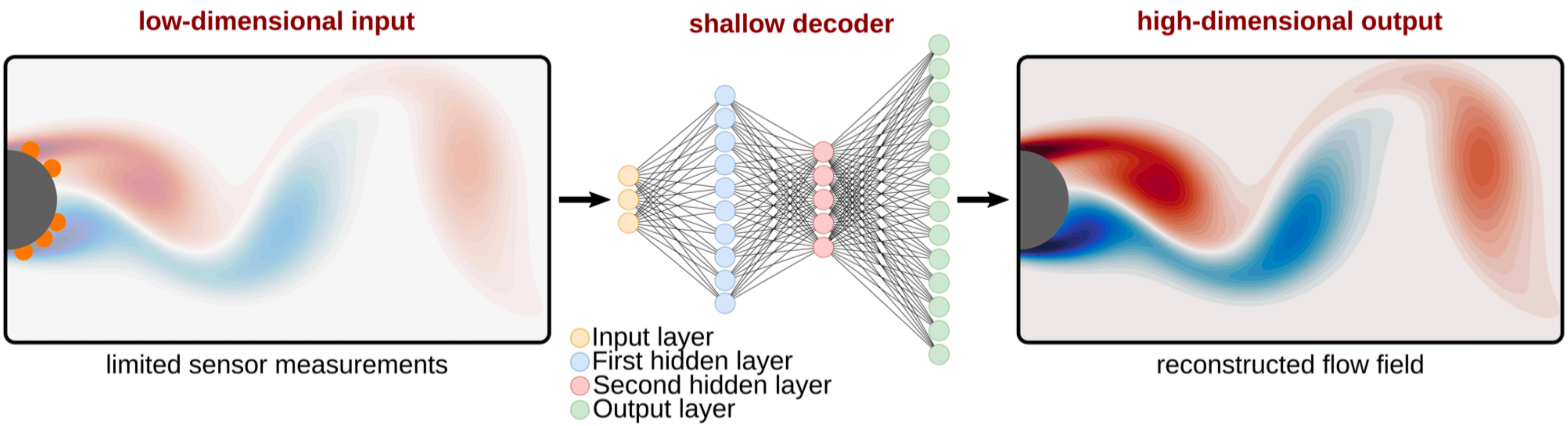
Nonlinear Mapping

General Form: Compositional Layers

$$\mathcal{F}(\boldsymbol{s};\mathbf{W}) := R(\mathbf{W}^K R(\mathbf{W}^{K-1} \cdots R(\mathbf{W}^1 \boldsymbol{s})))$$

Universal Approximators: Hornik 1990

Structure of Mapping



Linear Maps: SVD (left singular vector) defines layer

Shallow Layer Mapping

Two Layers

$$\mathcal{F}(\boldsymbol{s}) \;=\; \boldsymbol{\Omega}(\boldsymbol{\nu}(\boldsymbol{\psi}(\boldsymbol{s})))$$

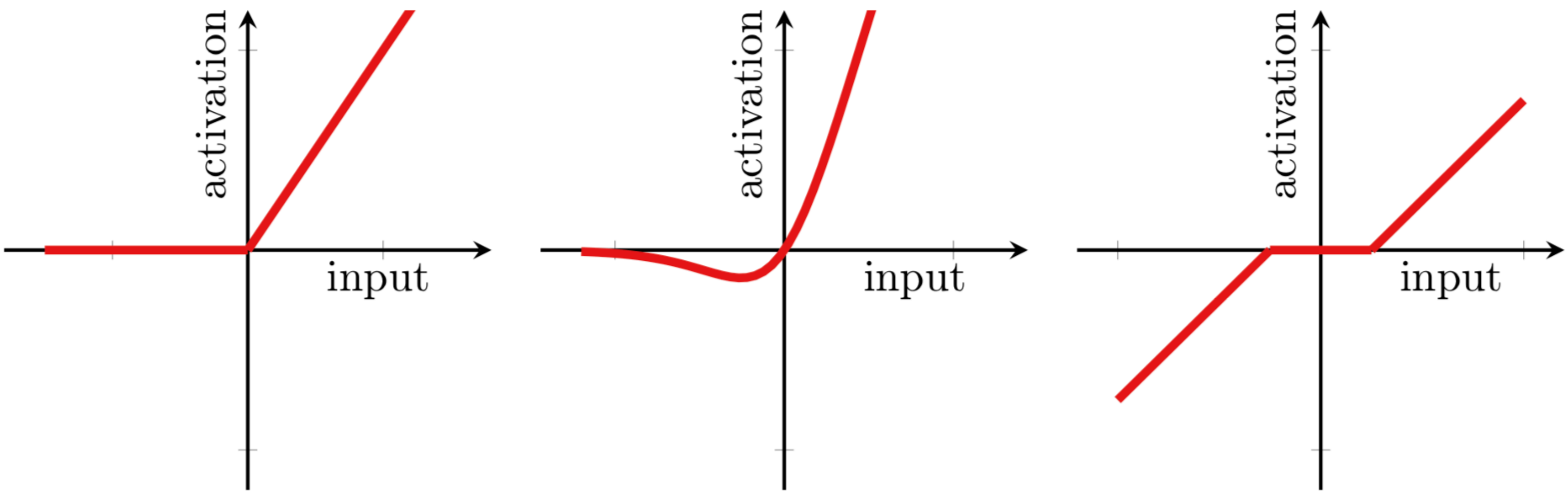
Composition

$$\boldsymbol{z}^{\boldsymbol{\psi}} = \boldsymbol{\psi}(\boldsymbol{s}) := R(\boldsymbol{W}^{\boldsymbol{\psi}}\boldsymbol{s} + \boldsymbol{b}^{\boldsymbol{\psi}}),$$

$$\boldsymbol{z}^{\boldsymbol{\nu}} = \boldsymbol{\nu}(\boldsymbol{z}^{\boldsymbol{\psi}}) := R(\boldsymbol{W}^{\boldsymbol{\nu}}\boldsymbol{z}^{\boldsymbol{\psi}} + \boldsymbol{b}^{\boldsymbol{\nu}})$$

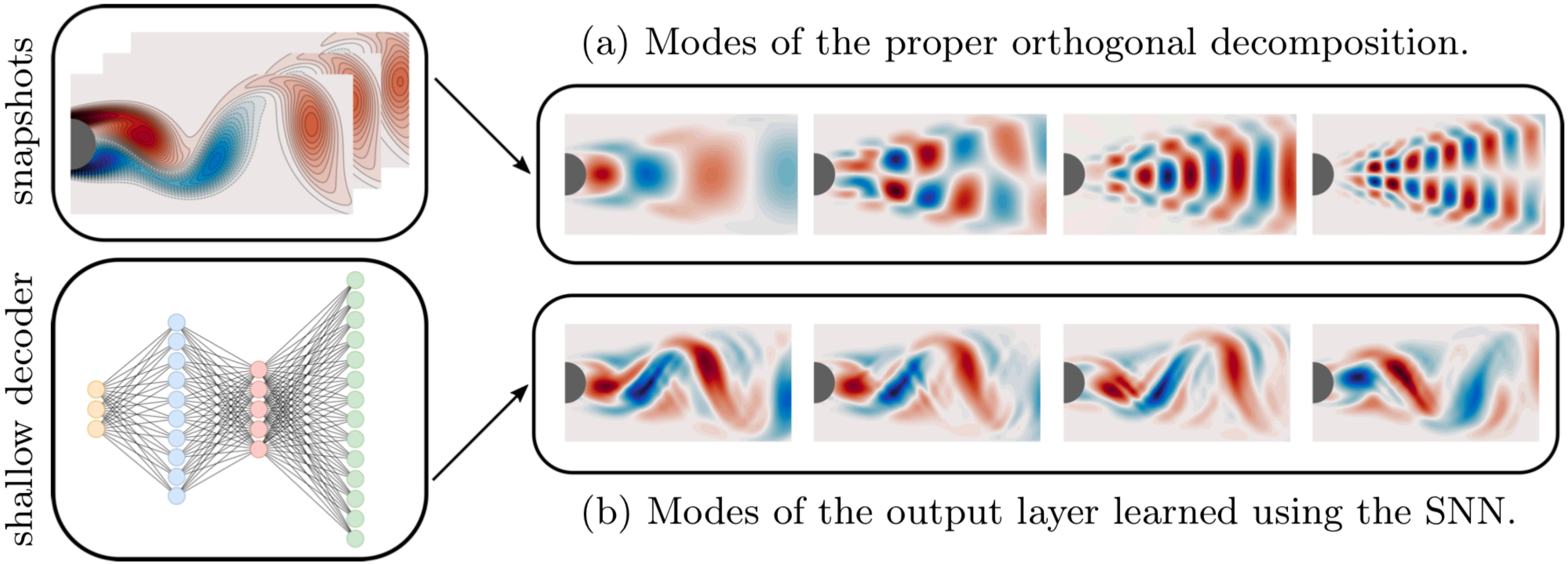
Erichson, Mathelin, Brunton, Mahoney & Kutz (2019)

Activation Functions



(a) ReLU (b) Swish (c) SoftShrinkage

Linear vs Nonlinear Maps

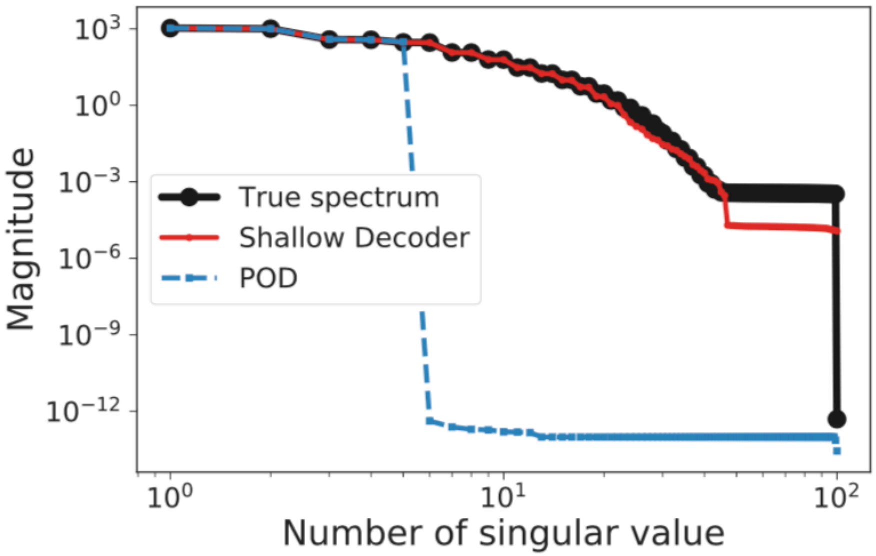


(a) Modes of the proper orthogonal decomposition.

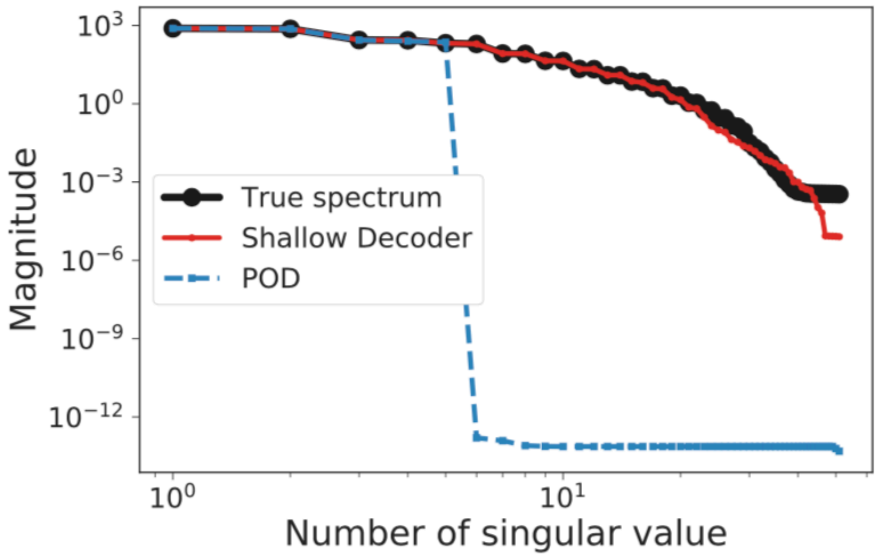
(b) Modes of the output layer learned using the SNN.

Improved Interpretability of Modes

Improved Performance

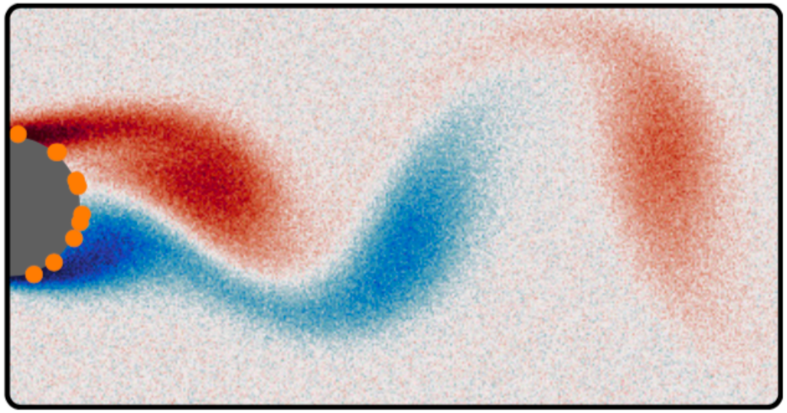


(a) Training data

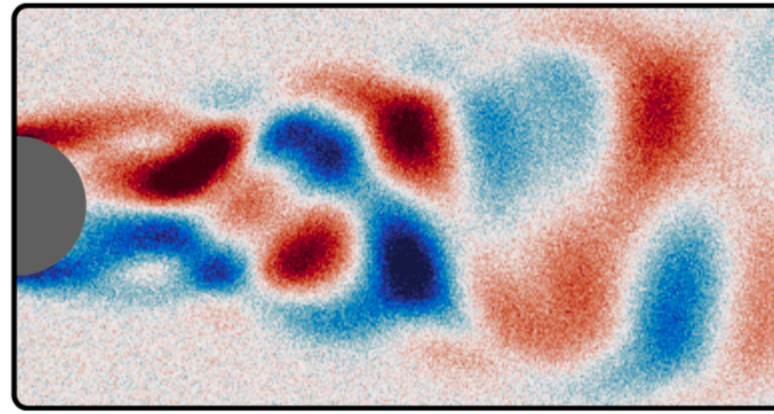


(b) Validation data

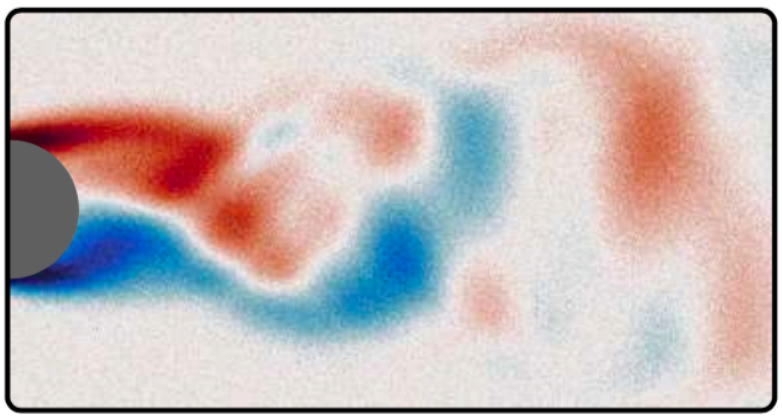
Robustness to Noise



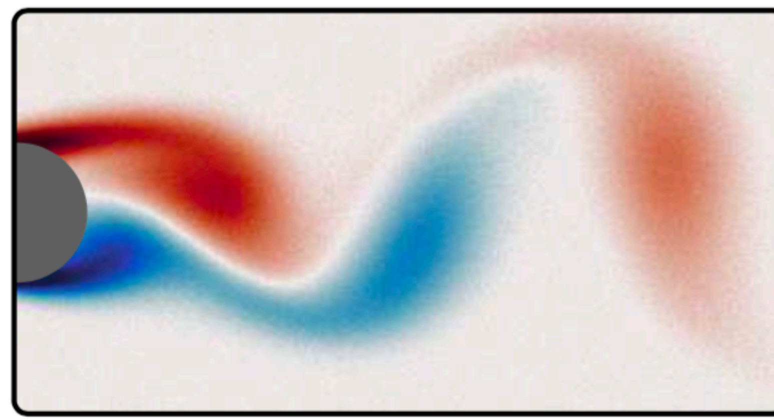
(a) Truth



(b) POD

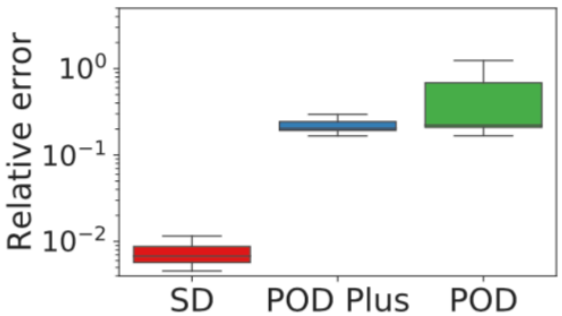


(c) POD Plus

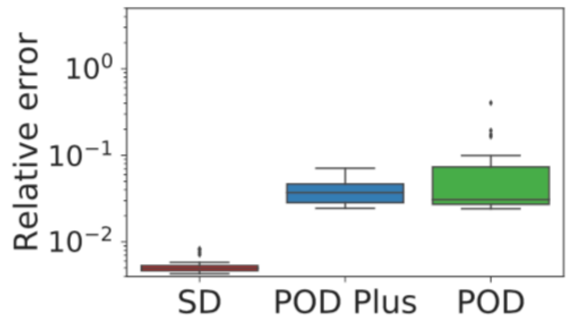


(d) Shallow Decoder

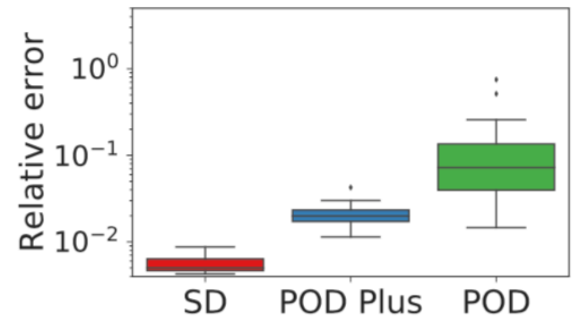
Comparison to Linear Methods



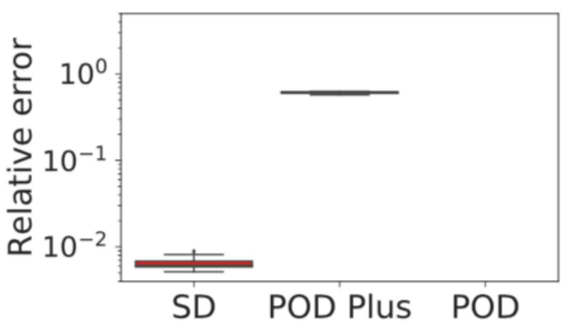
(a) 5 sensors.



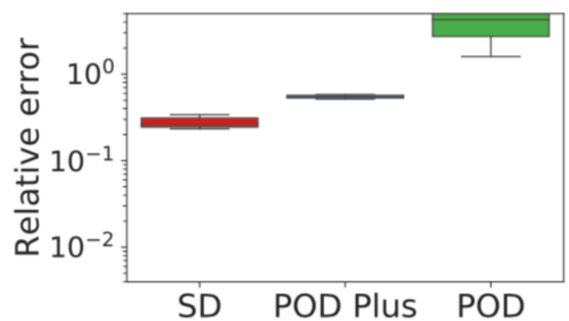
(b) 10 sensors.



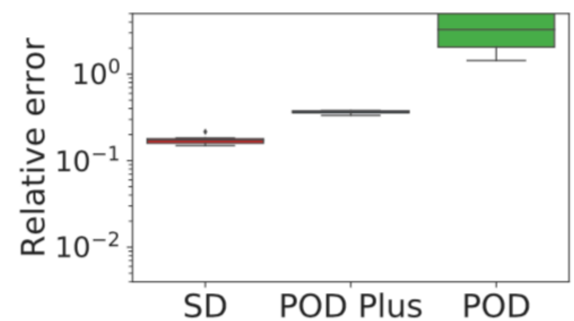
(c) 15 sensors.



(d) Nonlinear measurements.

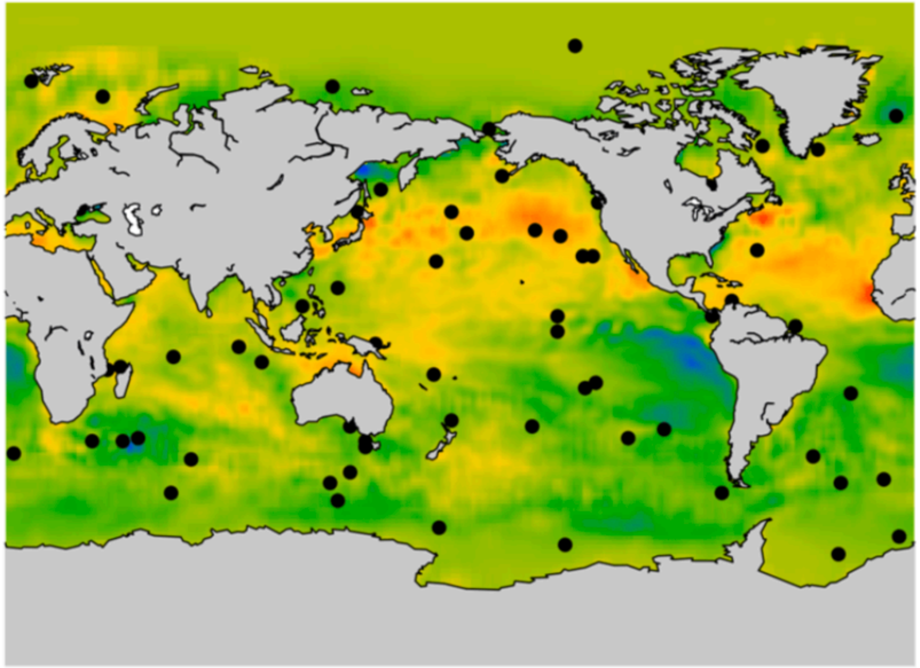


(e) SNR 10.

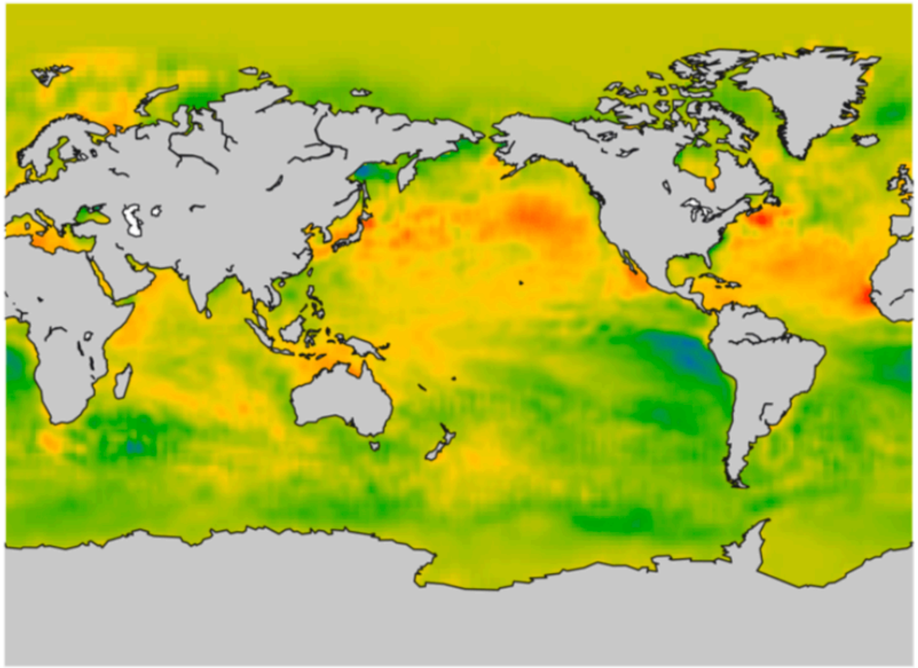


(f) SNR 50.

Improved Performance

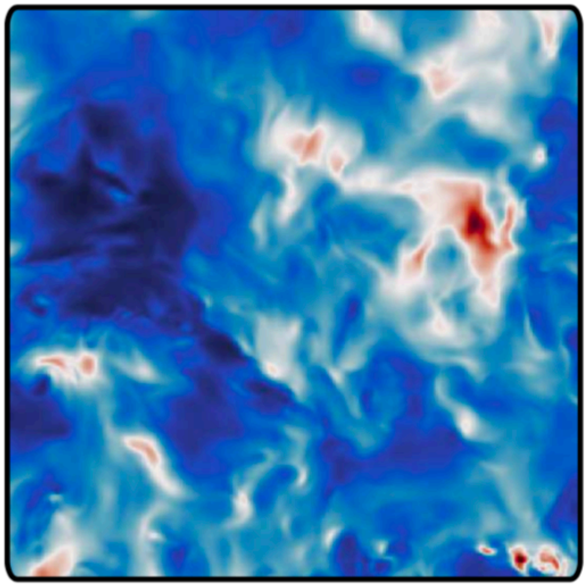


(a) Truth

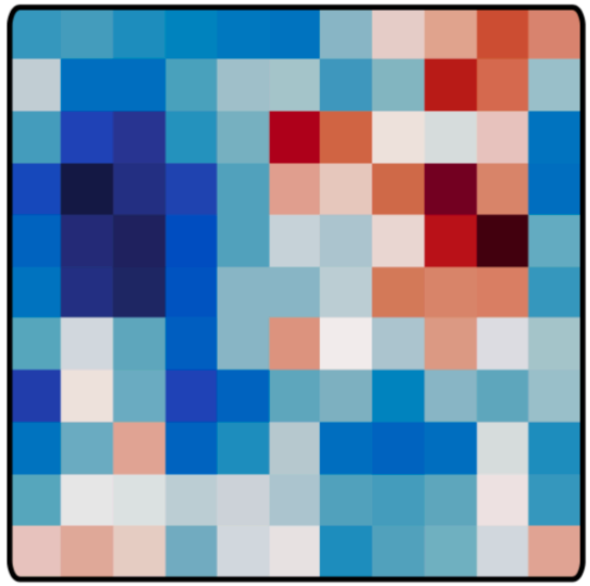


(b) Shallow Decoder

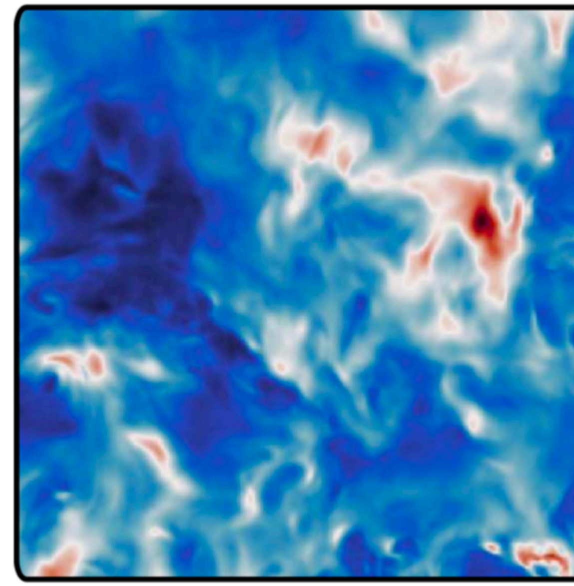
Super Resolution Analysis



(a) Snapshot



(b) Low resolution



(c) Shallow Decoder

Conclusion

Linear Measurements: Optimal via QR pivoting

Cost Constraints: Point measurements can be modified for a cost landscape

Nonlinear Measurements: Constructed via shallow decoder network

- Improved interpretability
- More robust to noise
- Allows for super resolution
- Significant reduction in training data

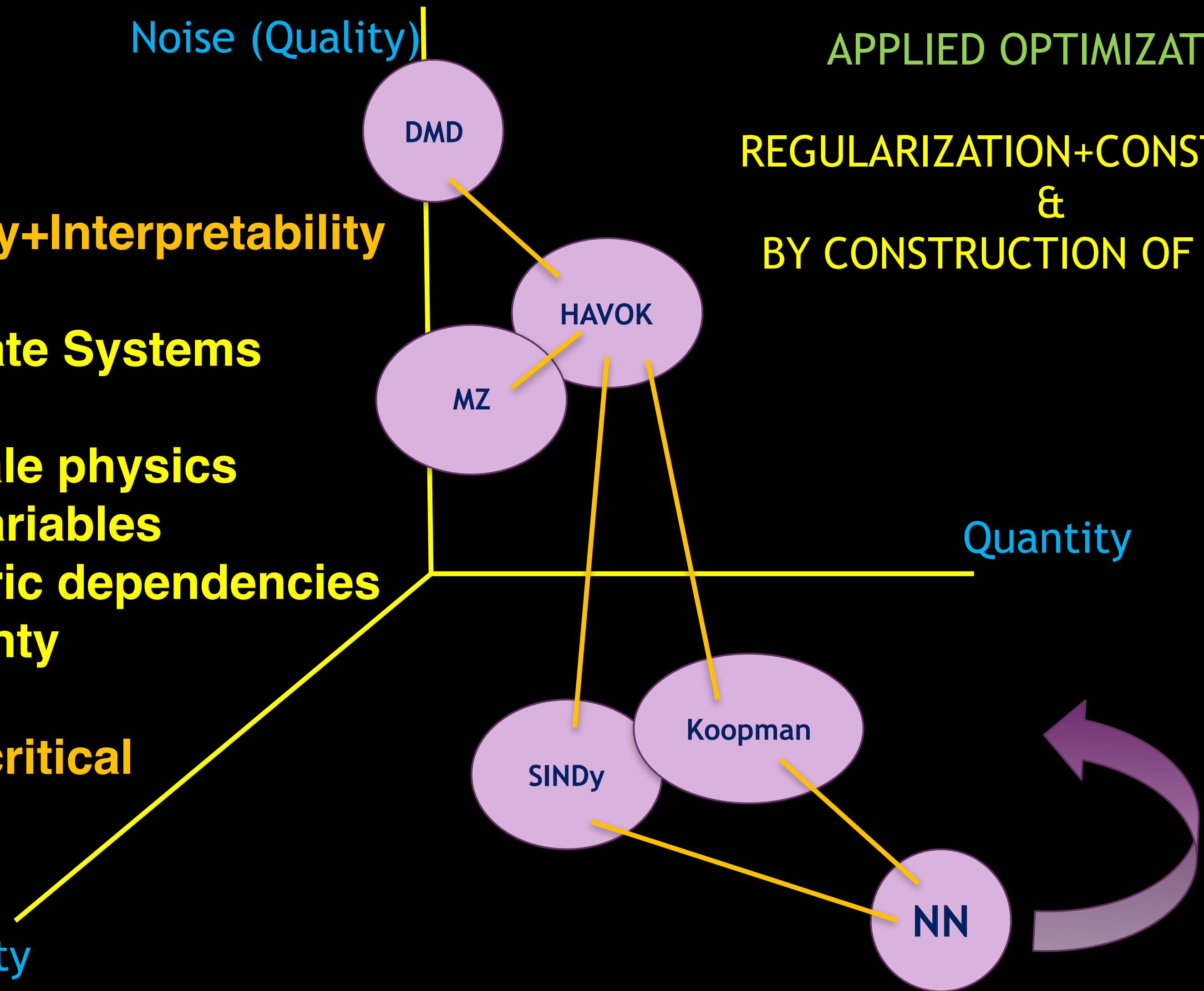
Erichson, Mathelin, Brunton, Mahoney & Kutz (2019)

A Diversity of Strategies

Generalizability+Interpretability

- Coordinate Systems
- Noise
- Multi-scale physics
- Latent variables
- Parametric dependencies
- Uncertainty

Parsimony is critical



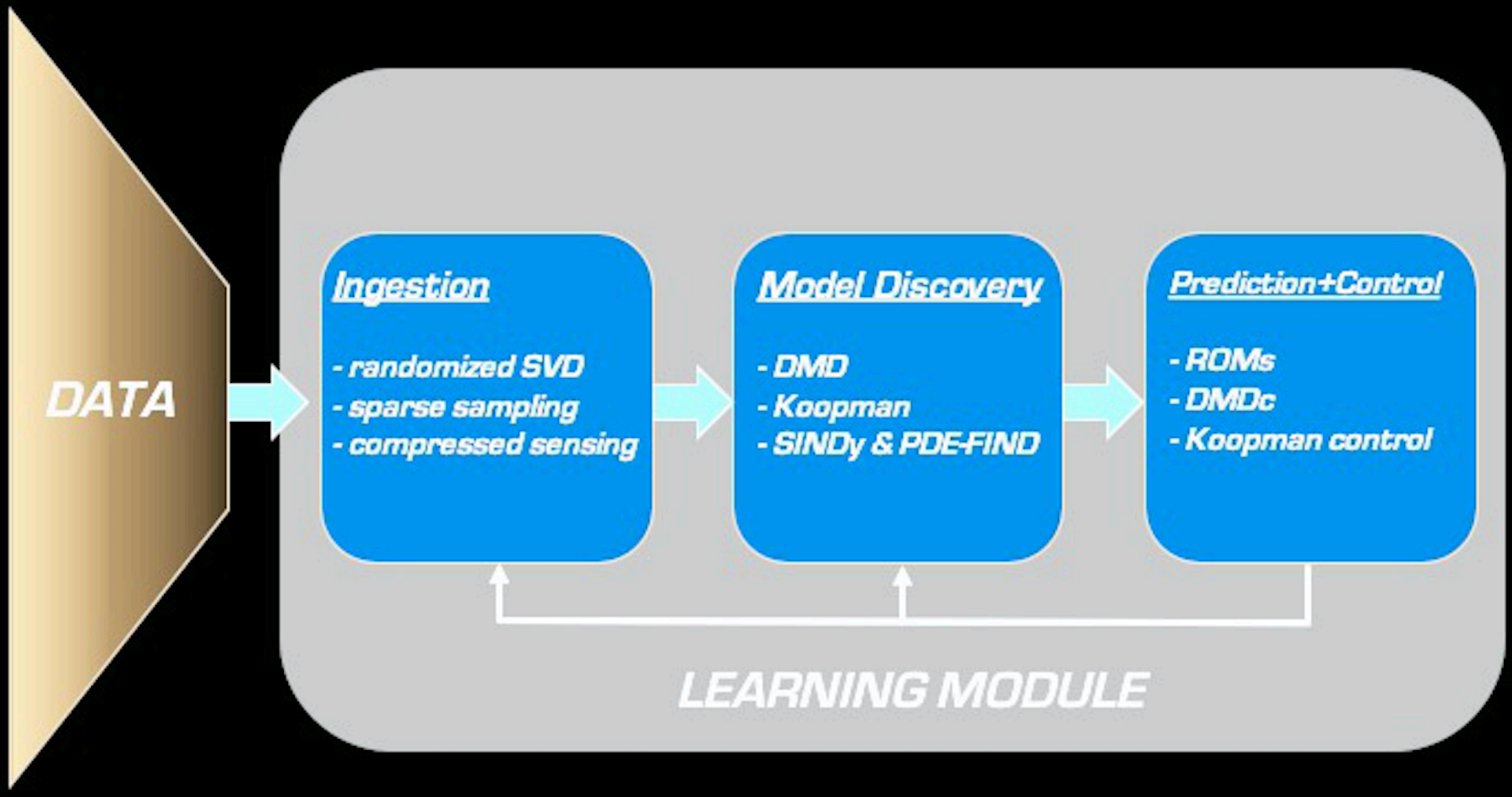
APPLIED OPTIMIZATION

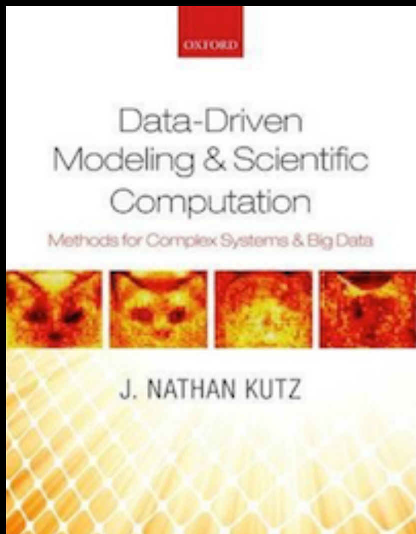
REGULARIZATION+CONSTRAINTS
&
BY CONSTRUCTION OF MODEL

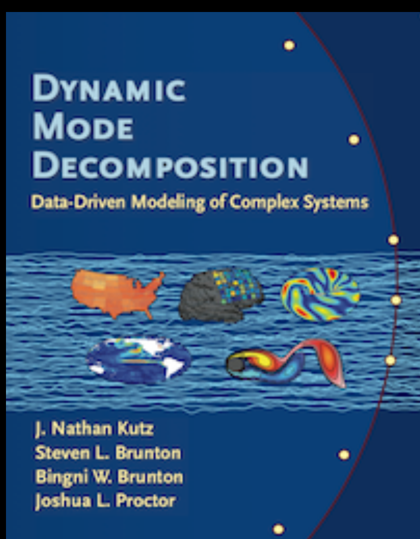
Conclusions

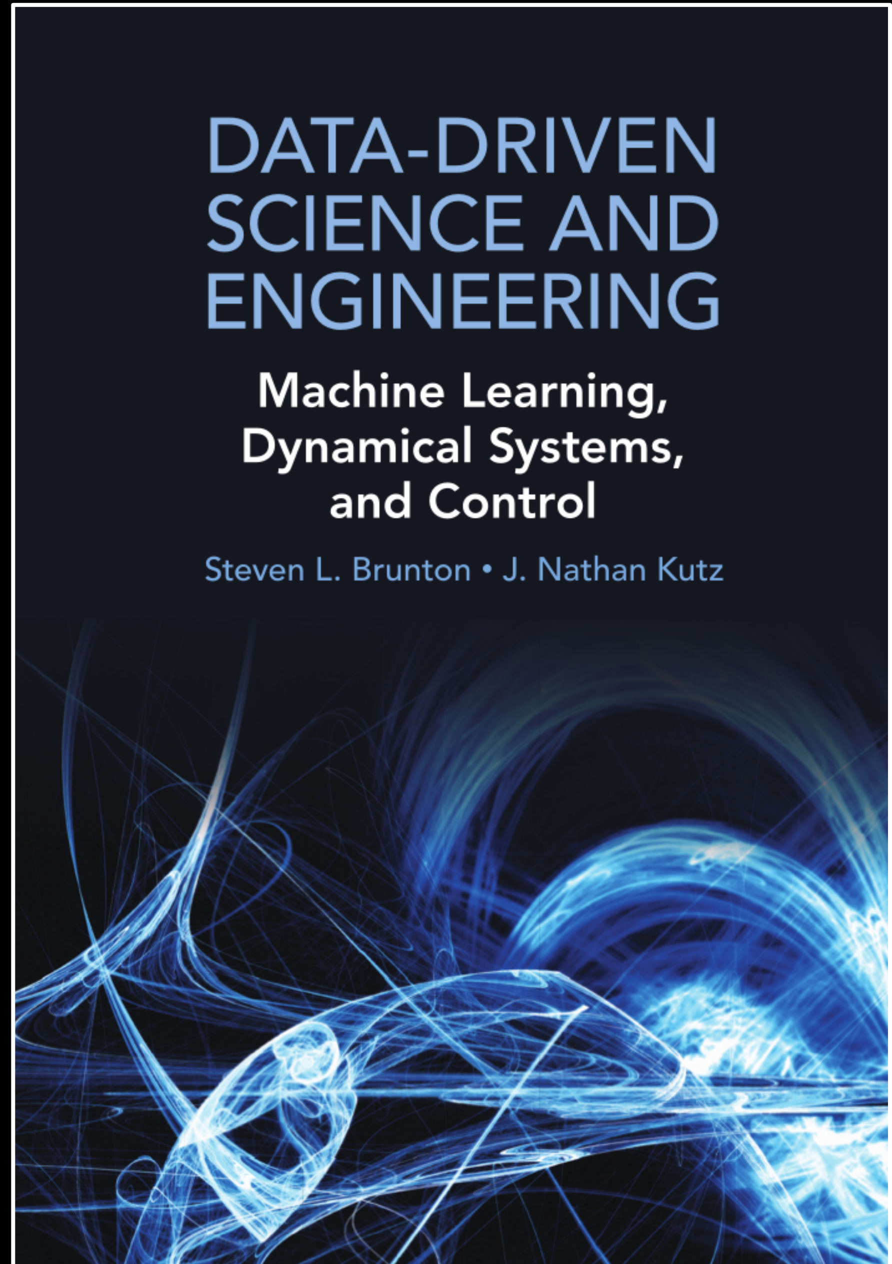
Model Selection & Sparse Regression Matter

- Principled approach to determining dynamics & coordinates
- (i) classification, (ii) reconstruction, (iii) future state prediction
- Sensors should be maximally informative









YouTube Resources & Open Source Code