Noise robustness: weak & ensemble SINDy

CWI Autumn School - Scientific Machine Learning and Dynamical Systems

Urban Fasel

Imperial College London

Literature

- **Ensemble SINDy:**
  - U Fasel, JN Kutz, BW Brunton, SL Brunton (2022) Ensemble-SINDy: Robust sparse model discovery in the low-data, high-noise limit, with active learning and control.
  - SM Hirsh, DA Barajas-Solano, JN Kutz (2022) Sparsifying priors for Bayesian uncertainty quantification in model discovery.
  - LM Gao, U Fasel, SL Brunton, JN Kutz (2023) Convergence of uncertainty estimates in Ensemble and Bayesian sparse model discovery.

- **Integral and weak form SINDy:**
  - H Schaeffer, SG McCalla (2017) Sparse model selection via integral terms.
  - PAK Reinbold, DR Gurevich, RO Grigoriev (2020) Using noisy or incomplete data to discover models of spatiotemporal dynamics.
  - D Messenger, DM Bortz (2021) Weak SINDy for partial differential equations.
  - AA Kaptanoglu, …, ZG Nicolaou, … (2022) **PySINDy**, SINDyCP

Part 4 outline

- **Ensemble SINDy**
  - Ensemble methods for noise robust model identification
    - Bootstrap aggregating the time series data or library terms
  - Active learning
  - UQ-SINDy
  - MATLAB example

- **Weak-SINDy**
  - Integral and weak form SINDy methods
  - Main concept and implementation
    - MATLAB example parameter estimation
    - Additional MATLAB and Python tutorials weak-SINDy

Ensemble SINDy

Ensemble SINDy

Combining predictions from multiple models

→ Improve accuracy of model prediction

→ Reduce variance and help avoid overfitting

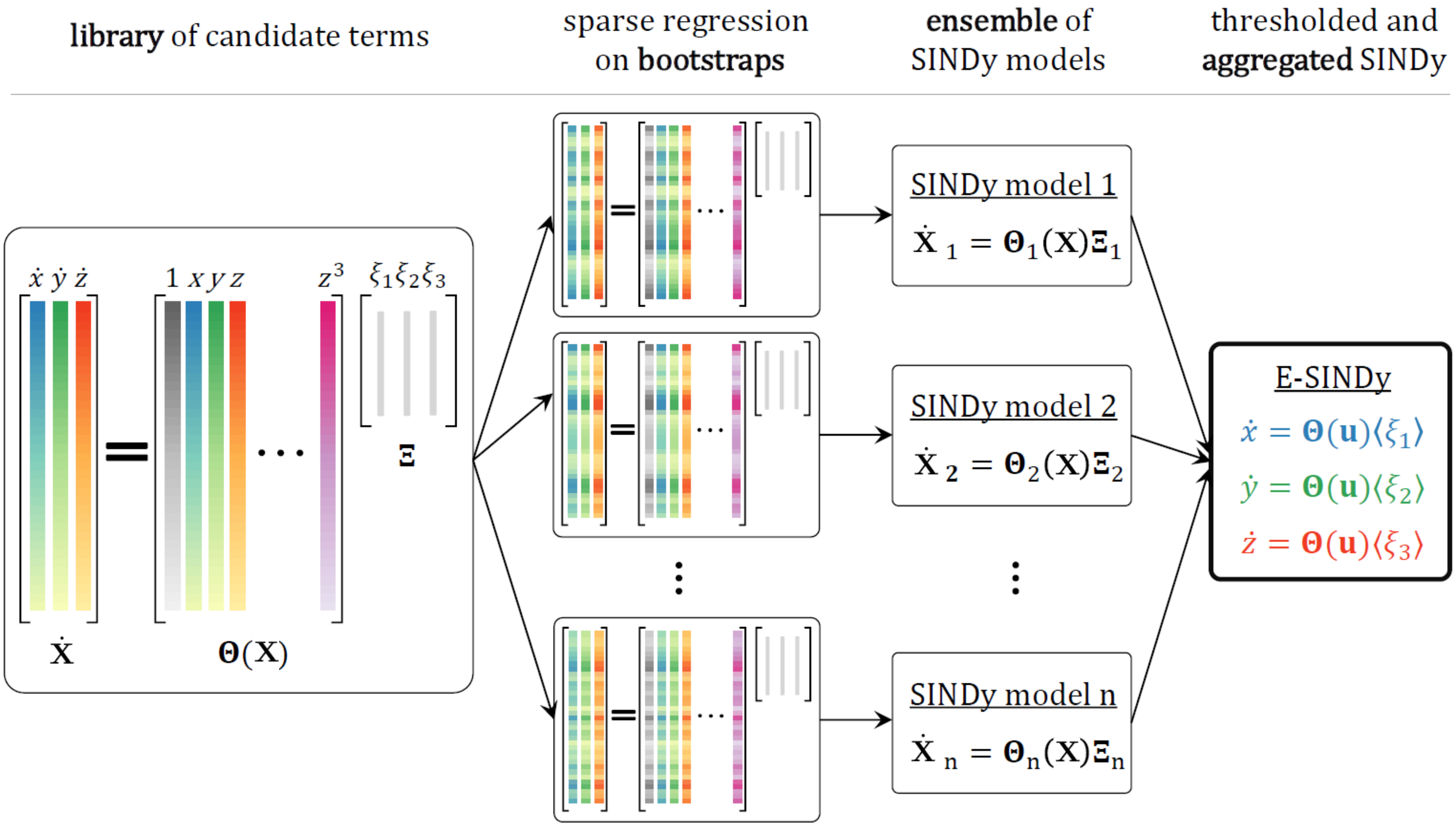
Popular ensemble learning method

→ Bootstrap aggregating or bagging

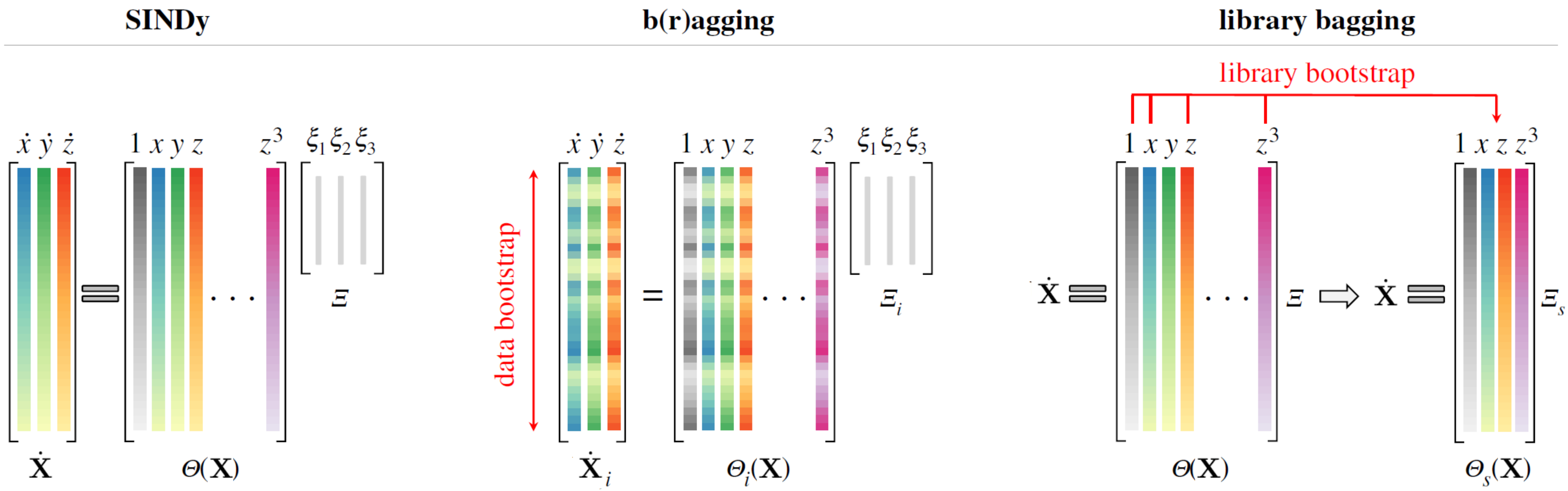
Connections to model selection

→ Cross-validation and stability selection use ensemble of models to select best model

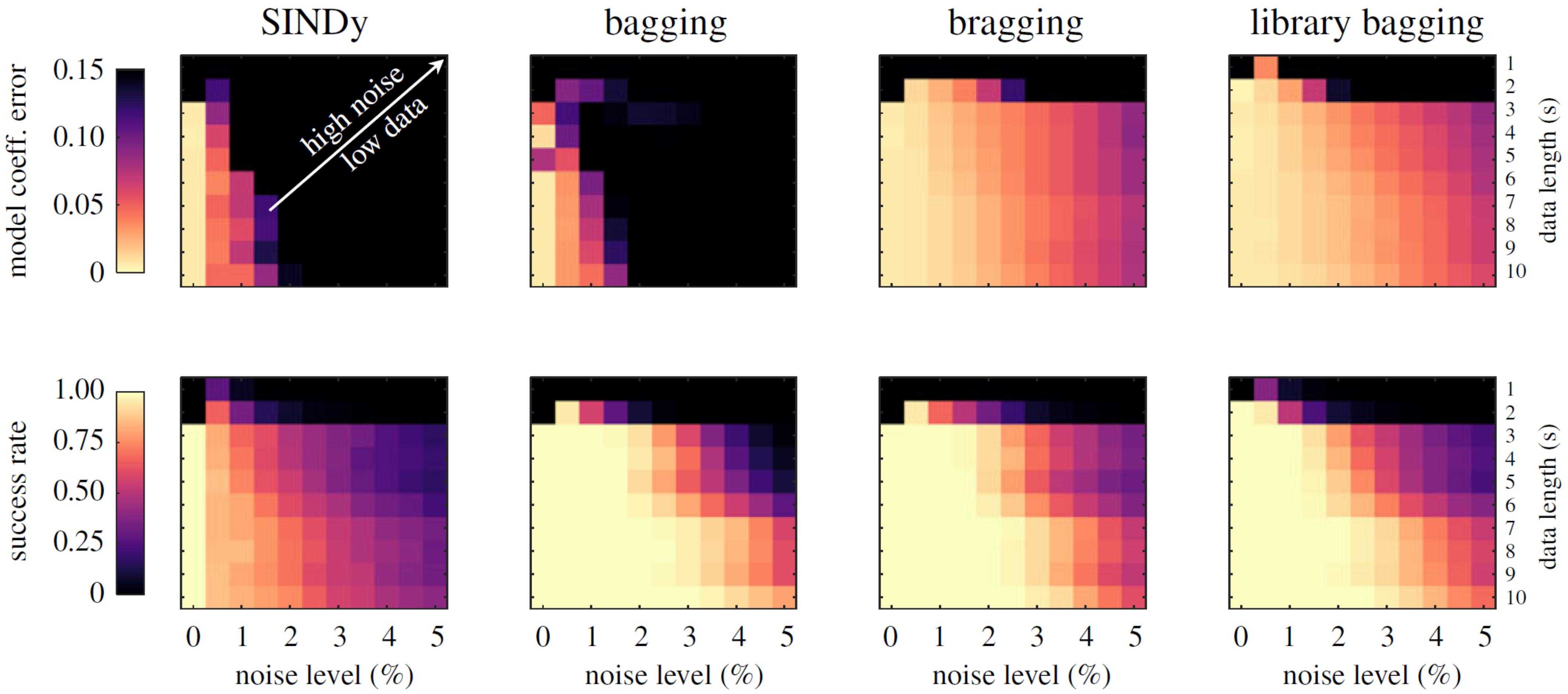
Ensemble SINDy



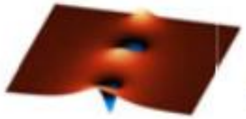
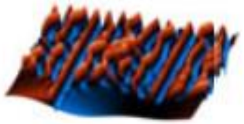
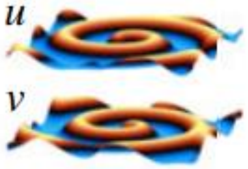
Ensemble SINDy



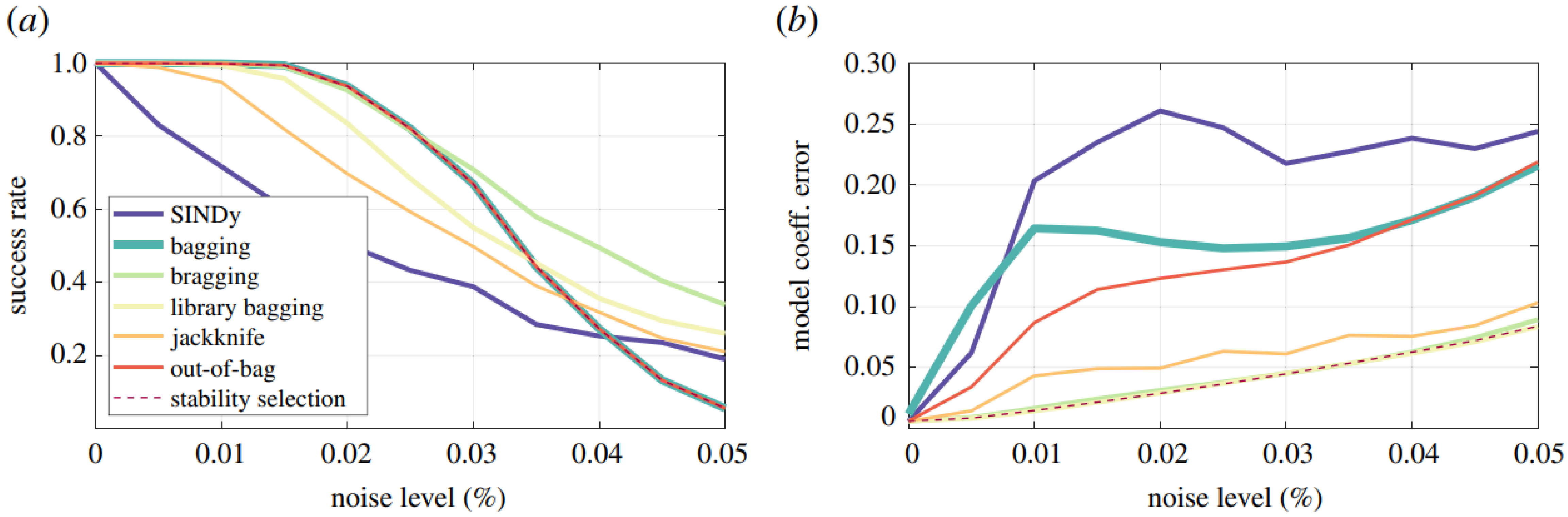
Ensemble SINDy



Ensemble SINDy

PDE		noise level		form	model error WSINDy	model error E-WSINDy	success rate WSINDy	success rate E-WSINDy
	inviscid Burgers		100%	$u_t + 0.5uu_x = 0$	2.6%	**2.5%**	99%	**100%**
	Korteweg de Vries		100%	$u_t + 0.5uu_x + u_{xxx} = 0$	27.5%	**4.0%**	93.5%	**100%**
	nonlinear Schrödinger		50%	$iu_t + 0.5u_{xx} + \lvert u\rvert^2 u = 0$	13.0%	**11.3%**	82.0%	**100%**
	Kuramoto–Sivashinsky		100%	$u_t + 0.5uu_x + u_{xx} + u_{xxxx} = 0$	29.5%	**24.7%**	87.5%	**99.5%**
u, v 	reaction–diffusion		20%	$u_t = 0.1\nabla^2 u + \lambda(A)u - \omega(A)v$ $v_t = 0.1\nabla^2 v + \omega(A)u + \lambda(A)v$ $A^2 = u^2 + v^2,\ \omega = -\beta A^2,\ \lambda = 1 - A^2$	77.7%	**7.1%**	0.0%	**99.5%**

Ensemble SINDy – different sampling strategies



(a) Success rate: identifying correct model structure

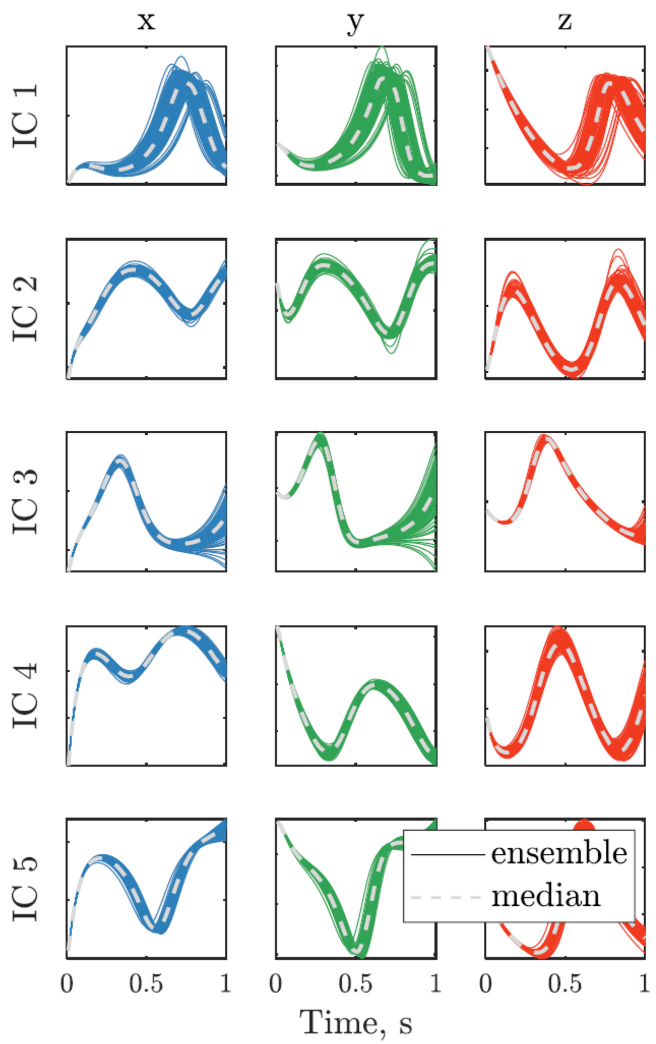
(b) model coefficient error

Bragging (robust bagging) SINDy generally performs best

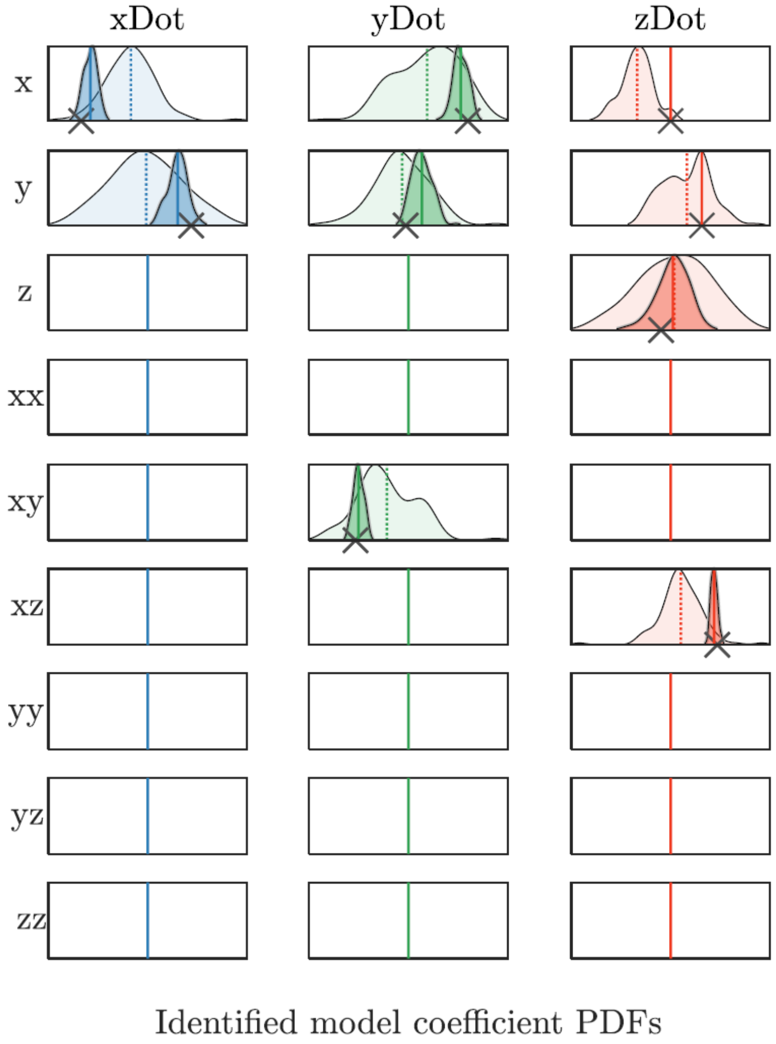
- <u>Robust bagging</u>: aggregate by taking median of identified models

Ensemble SINDy – active learning

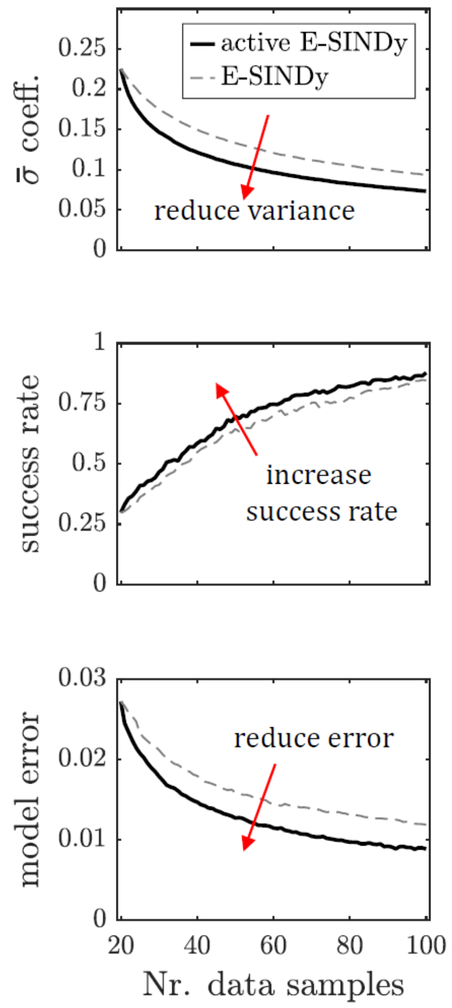
Exploiting ensemble statistics
for **active learning**



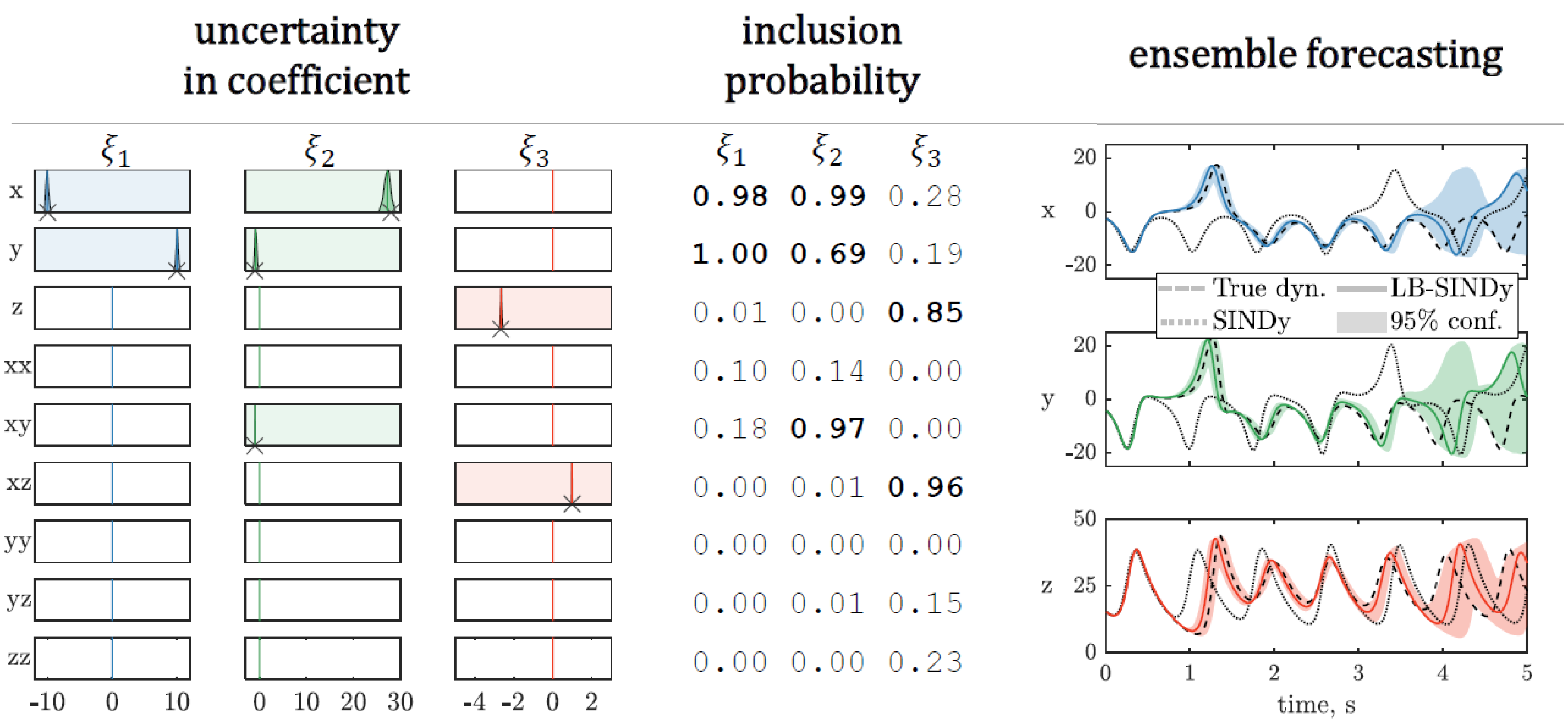
Reducing variance
of model coefficient



Improving **data-efficiency**
of model discovery



Ensemble SINDy – probabilistic models



- **Robust model discovery**: high noise, low data limit
- E-SINDy identifies **probabilistic models** at low computational cost

Ensemble SINDy tutorials

- **MATLAB**
  - **E-SINDy vs SINDy**: Identify Lorenz system on noisy data
  - GitHub RSPA paper: https://github.com/urban-fasel/EnsembleSINDy
    - ODE and PDE examples
    - Active learning
    - MPC

- **PySINDy**
  - https://github.com/dynamicslab/pysindy/blob/master/examples/13_ensembling.ipynb
  - ODE and PDE examples

UQ-SINDy – sparse Bayesian inference

Estimating model coefficient distributions

→ model coefficient uncertainty, due to observation errors, limited data

→ model coefficient inclusion probability: how likely a coefficient is active

Bayes rule: $p(\boldsymbol{\Xi}\,|\mathbf{X}) \propto p(\mathbf{X}|\boldsymbol{\Xi}\,)p(\boldsymbol{\Xi})p(\boldsymbol{x}_0)$

- $p(\boldsymbol{\Xi}\,|\mathbf{X})$ **posterior** distribution of the model coefficients $\boldsymbol{\Xi}$ conditioned on the data $\mathbf{X}$
- $p(\mathbf{X}|\boldsymbol{\Xi})$ **likelihood** of the model with coefficient $\boldsymbol{\Xi}$ given the data
- $p(\boldsymbol{\Xi})$ **prior** distribution of the model coefficients

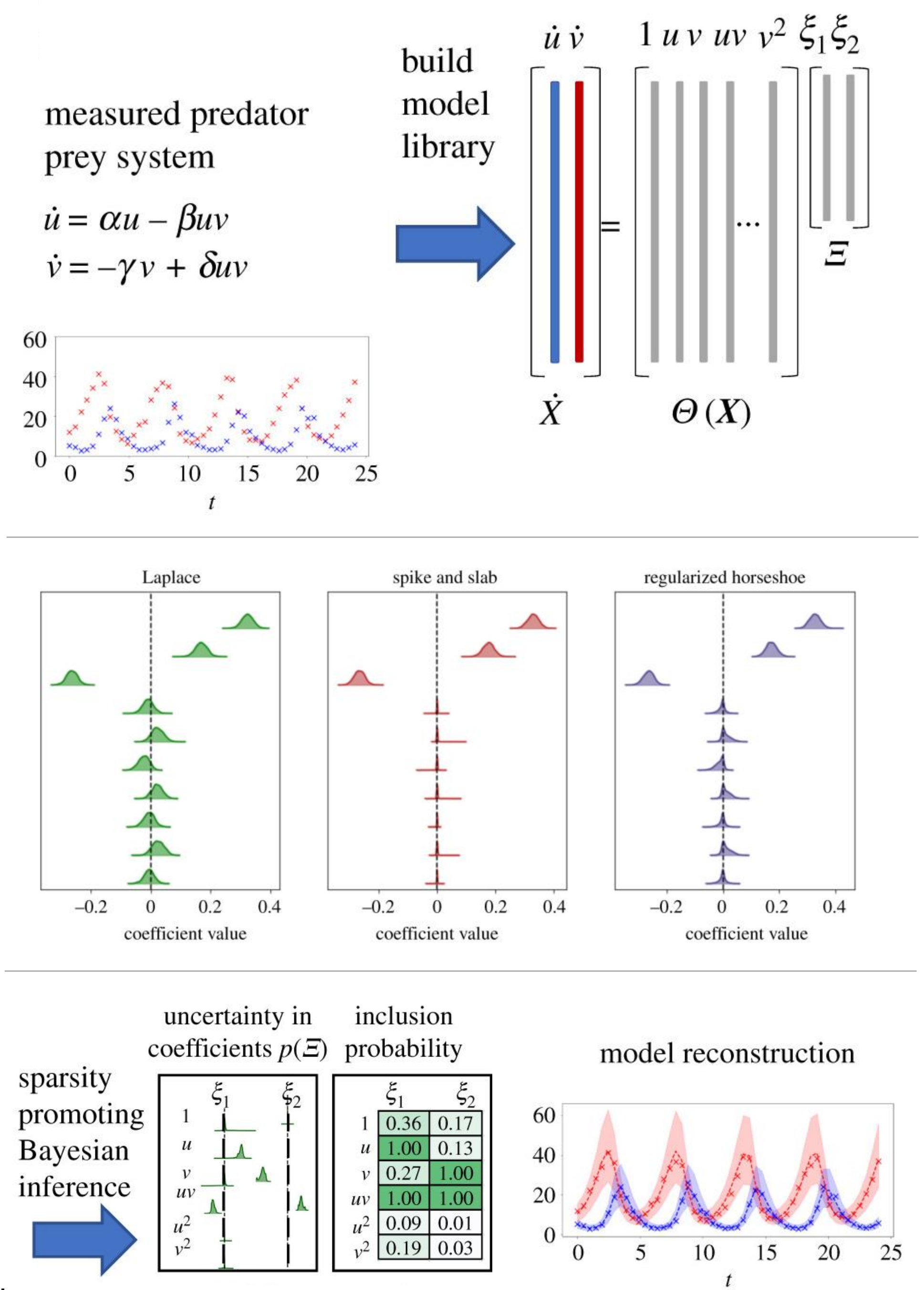
We want to **learn the posterior distribution** of the model coefficients $\boldsymbol{\Xi}$

- Computing posterior distribution not analytically tractable
- Sampling-based methods such as Markov chain Monte Carlo (MCMC)

Paper: SM Hirsh, DA Barajas-Solano, JN Kutz (2022) [Sparsifying priors for Bayesian uncertainty quantification in model discovery].

UQ-SINDy – method

1. Collect **data**
2. Construct **library**
   - Symbolic library → used as generative model
3. Construct model **likelihood**
   - Lognormal likelihood: $p(\mathbf{X}|\boldsymbol{\Xi}) = \prod\prod \frac{1}{\sigma_j\sqrt{2\pi}} \exp\left[\frac{-1}{2\sigma_j^2}\left|\log y_{i,j} - \log \hat{x}_j\left(t_j, \boldsymbol{\Xi}\right)\right|^2\right]$
   - SINDy prediction at time t for model $\boldsymbol{\Xi}$: $\hat{x}(t, \boldsymbol{\Xi}) = \boldsymbol{x}_0 + \int_{t_0}^{t} \boldsymbol{\Theta}\big(\boldsymbol{x}(t')\big)\boldsymbol{\Xi}\mathrm{d}t'$
4. Construct model **priors**
   1. <u>Sparsity promoting priors</u>: 1) spike and slab, 2) regularized horseshoe
      - e.g. Hierarchical spike and slab prior: $\xi_j|\lambda_j \sim \mathcal{N}(0, c^2)\lambda_i, \quad \lambda_j \sim \mathrm{Ber}(\pi)$
      - If $\lambda = 1$: ξ_j follows slab distribution ($\mathcal{N}(0, c^2)$)
      - if $\lambda = 0$: ξ_j follows spike distribution (Dirac delta distribution centered at zero)
5. Bayesian **inference**
   - Employ MCMC to draw samples from the posterior distribution
     - Sample SINDy models and integrate over t to compute $\hat{x}(t, \boldsymbol{\Xi})$
     - **Challenge**: integrating potentially unstable models



Paper: SM Hirsh, DA Barajas-Solano, JN Kutz (2022) Sparsifying priors for Bayesian uncertainty quantification in model discovery.

UQ-SINDy – sparse Bayesian inference

Python code GitHub paper Seth Hirsh Python

→ https://github.com/sethhirsh/BayesianSindy/blob/master/figures/figure3.ipynb

Alternative UQ-SINDy papers

→ S Zhang, G Lin (2018) Robust data-driven discovery of governing physical laws with error bars.

→ Y Yang, MA Bhouri, P Perdikaris (2020) Bayesian differential programming for robust systems identification under uncertainty.

Paper with code, introducing sparsity promoting priors

→ D Korobilis, K Shimizu (2021) Bayesian Approaches to Shrinkage and Sparse Estimation.

→ GitHub https://github.com/korobilis/hierarchicalbayes → Code\Linear Regression\SkinnyGibbs

Tutorial Bayesian Inference to fit parameters of dynamical system

→ S Jbabdi https://users.fmrib.ox.ac.uk/~saad/ONBI/bayes_practical.html

E-SINDy and UQ-SINDy connections

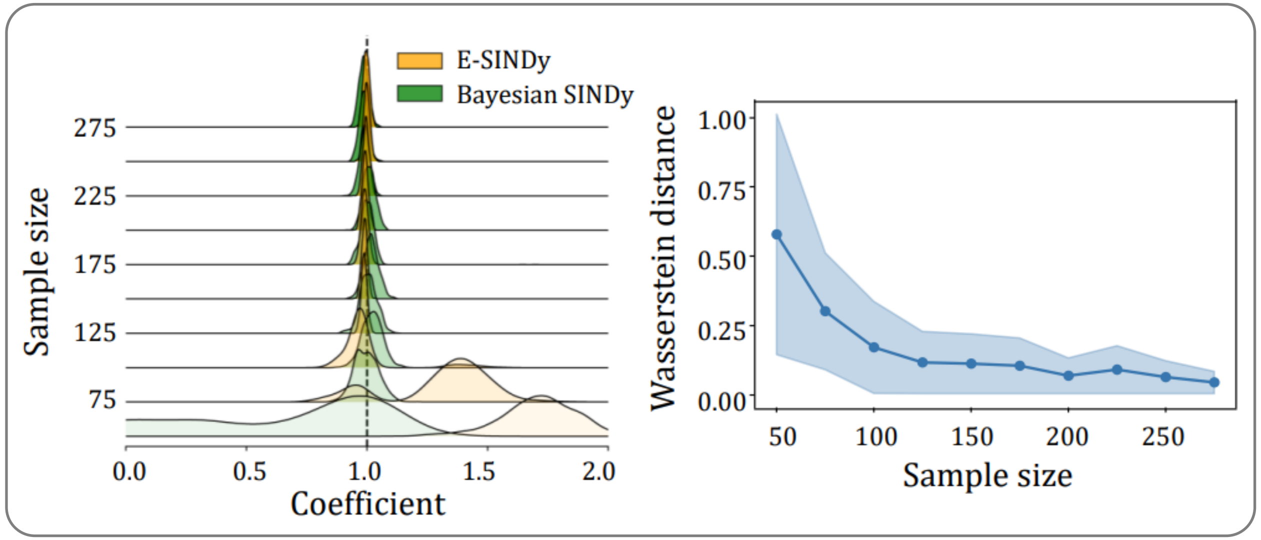
Ensemble SINDy …

→ is asymptotically equivalent to Bayesian-SINDy

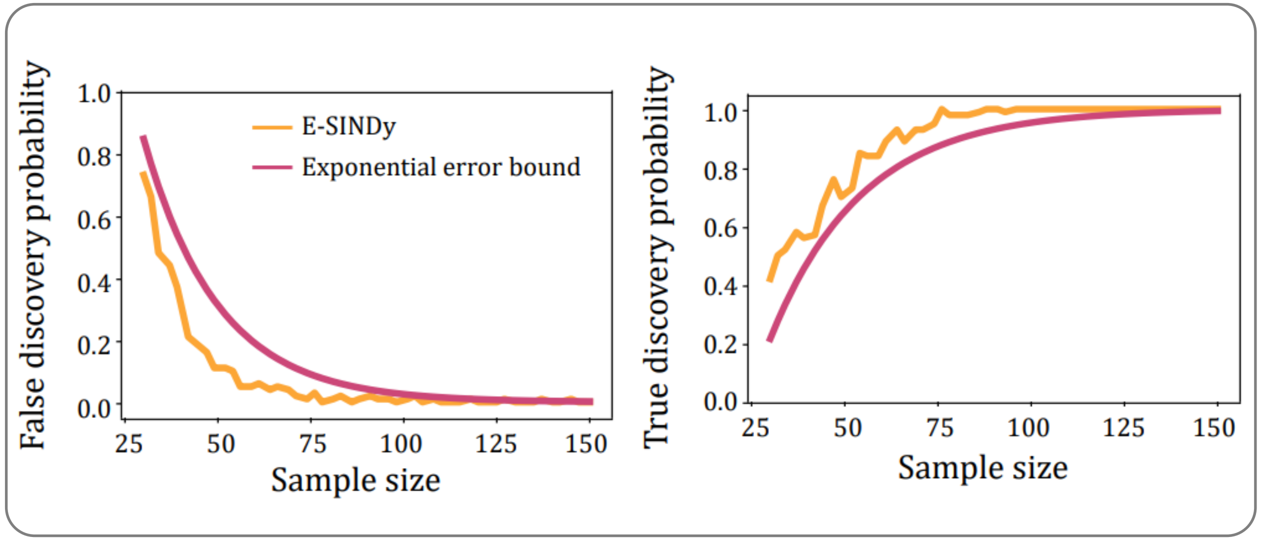
→ has lower computational cost → ≈3000 times less expensive

→ can perform valid uncertainty quantification with statistical guarantee

(a) Uncertainty quantification guarantees



(b) Exponential convergence theorem



Paper: LM Gao, U Fasel, SL Brunton, JN Kutz (2023) Convergence of uncertainty estimates in Ensemble and Bayesian sparse model discovery.

Weak SINDy

Weak SINDy

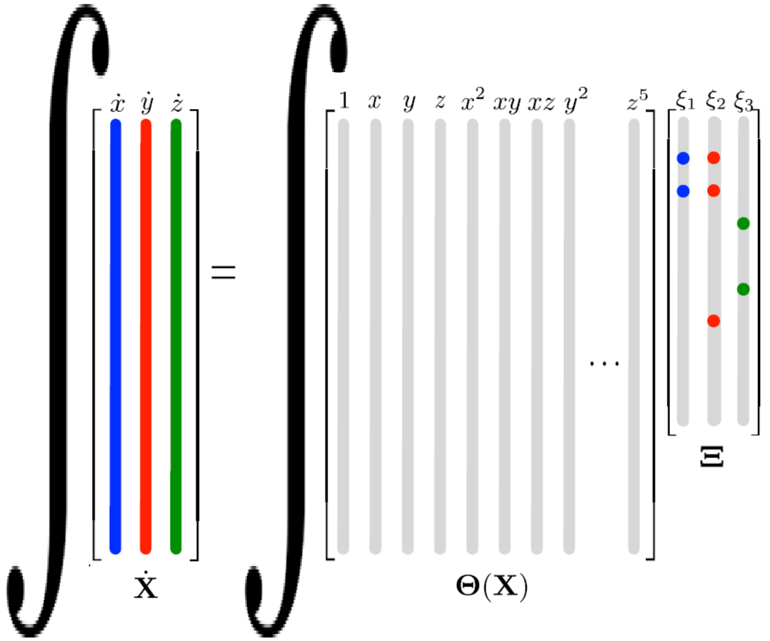
Integral formulation: H Schaeffer, SG McCalla (2017) [Sparse model selection via integral terms](#).

- Reformulate $\dot{x}_i = \Theta(\boldsymbol{x})\boldsymbol{\xi}_i = \sum f_j(\boldsymbol{x})\boldsymbol{\xi}_{i,j}$ to integral form

  $\rightarrow x_i(t) = x_i(0) + \sum d_j(\boldsymbol{x}, t)\xi_{i,j}$

  - $d_j(x,t) = \int_0^t f_j\big(\boldsymbol{x}(\tau)\big)d\tau$ integrated library function
  - $f_j(\boldsymbol{x})$ library function
  - $\xi_{i,j}$ model coefficient

- Solve ℓ_0-penalized least squares → same as standard SINDy



Weak form SINDy: generalizes integral form SINDy

- Integration against test function
- Moving derivatives off of the data onto test functions via integration by parts
- Different implementations
  - PAK Reinbold, DR Gurevich, RO Grigoriev (2020) [Using noisy or incomplete data to discover models of spatiotemporal dynamics](#).
  - D Messenger, DM Bortz (2021) [Weak SINDy for partial differential equations](#).
  - AA Kaptanoglu, …, ZG Nicolaou, … (2022) **[PySINDy](#)**, [SINDyCP](#)

Weak SINDy

Idea: Move derivatives off of the data onto a test functions via integration by parts

- PDE: $u_t = \boldsymbol{\Theta}(u)\boldsymbol{\xi} = \boldsymbol{\Theta}(u, u_x, u_{xx}, \dots, x)\boldsymbol{\xi}$
  - $u(x,t)$: state, $\boldsymbol{\Theta}$: library, $\boldsymbol{\xi}$: coefficients x: space, t: time
- Test function: $\phi(x,t)$ smooth & compactly supported: $\phi(x,t)$ *vanishes along domain bound*
  - e.g. $\phi(x,t) = (x^2-1)^p(t^2-1)^q$ on domain $\Omega = \{(\mathrm{x},\mathrm{t}) : |\mathrm{x}| \leq 1,\ |\mathrm{t}| \leq 1\}$

Integration by parts: remember product rule $\big(f(x)g(x)\big)' = f'(x)g(x) + f(x)g'(x)$

- $\int_a^b f(x)g'(x)dx = \int_a^b \big(f(x)g(x)\big)'dx - \int_a^b f'(x)g(x)dx = f(b)g(b) - f(a)g(a) - \int_a^b f'(x)g(x)dx$

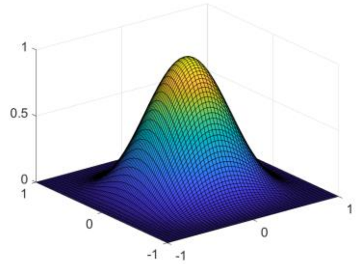
Weak form: Multiply each term in PDE with test function and integrate over k different domains Ω_k

- Domain Ω_k, e.g. 2D space x and time t, $d\Omega = dx\,dt$, $\partial\Omega_k$: domain bounds
- LHS: $\int_{\Omega_k} \phi u_t\, d\Omega = \cancelto{0}{[\phi(x,t)u(x,t)]_{\partial\Omega_k}} - \int_{\Omega_k} \phi_t u\, du = -\int_{\Omega_k} \phi_t u\, du$
- RHS: term by term integration by parts: move derivatives $u_x, u_{xx}, \ldots$ from data onto test function

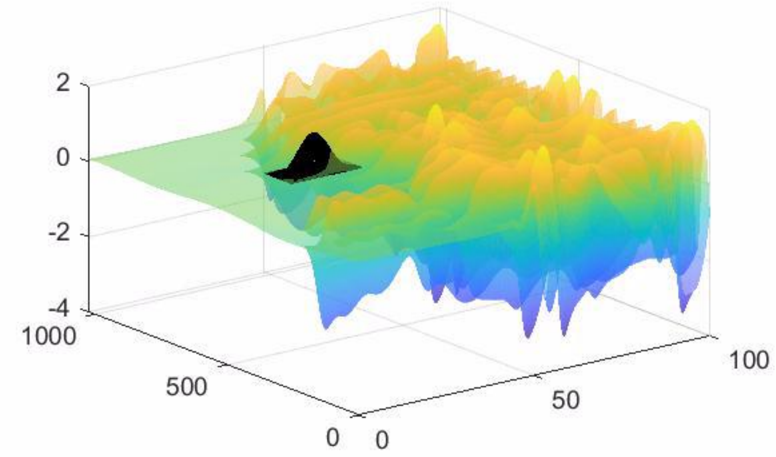
Weak form parameter estimation (SINDy on next slide)

Kuramoto-Sivashinsky equation: $u_t = -uu_x - u_{xx} - u_{xxxx}$

1. **Integration by parts**: multiply each term with test function and move derivative onto test function
   - Term u_t: $q_0^k = \int_{\Omega_k} \phi u_t \, d\Omega = -\int_{\Omega_k} \phi_t u \, d\Omega$
   - Term uu_x: $q_1^k = \int_{\Omega_k} \phi u \, u_x \, d\Omega = -\int_{\Omega_k} (\phi_x u + \phi u_x) \, u \, d\Omega = -\int_{\Omega_k} \phi_x u^2 \, d\Omega - \int_{\Omega_k} \phi u \, u_x \, d\Omega \quad \rightarrow \quad \int_{\Omega_k} \phi u u_x \, d\Omega = -\frac{1}{2} \int_{\Omega_k} \phi_x u^2 \, d\Omega$
   - Term u_{xx}: $q_2^k = \int_{\Omega_k} \phi u_{xx} \, d\Omega = -\int_{\Omega_k} \phi_x u_x \, d\Omega = \int_{\Omega_k} \phi_{xx} u \, d\Omega$
   - Term u_{xxxx}: $q_3^k = \int_{\Omega_k} \phi u_{xxxx} \, d\Omega = \int_{\Omega_k} \phi_{xxxx} u \, d\Omega$



2. **Test function**: $\phi = (\underline{x}^2 - 1)^p (\underline{t}^2 - 1)^q$
   - $p = 4,\ q = 3,\quad \underline{t} = (t - t_k)/H_t,\ \underline{x} = (x - x_k)/H_x$
3. **Integrate over domain**: $\Omega_k = \{(x, t) : |x - x_k| \le H_x, |t - t_k| \le H_t\}$
   - Integrate all q_i^k over N different domains Ω_k of size H_x and H_t



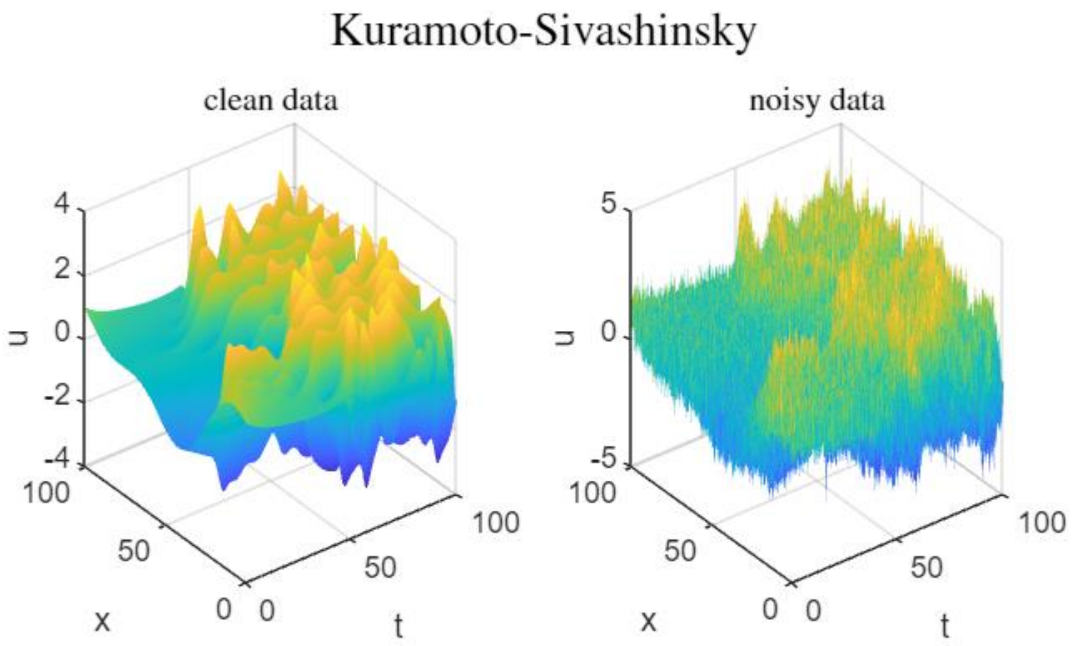
New system: $\boldsymbol{q}_0 = \sum_{n=1}^{3} \boldsymbol{q}_n \xi_n = \boldsymbol{Q}\boldsymbol{\xi} \qquad \boldsymbol{q}_0 \in \mathbb{R}^{\mathrm{k}},\ \boldsymbol{Q} \in \mathbb{R}^{\mathrm{k}\times 3},\ \boldsymbol{\xi} \in \mathbb{R}^3$

$\rightarrow \quad \hat{\boldsymbol{\xi}} = \operatorname{argmin}_{\boldsymbol{\xi}} \ \|\boldsymbol{q}_0 - \boldsymbol{Q}\boldsymbol{\xi}\|_2^2 \ + \ \lambda \|\boldsymbol{\xi}\|_0$

Weak form parameter estimation (SINDy on next slide)

MATLAB example

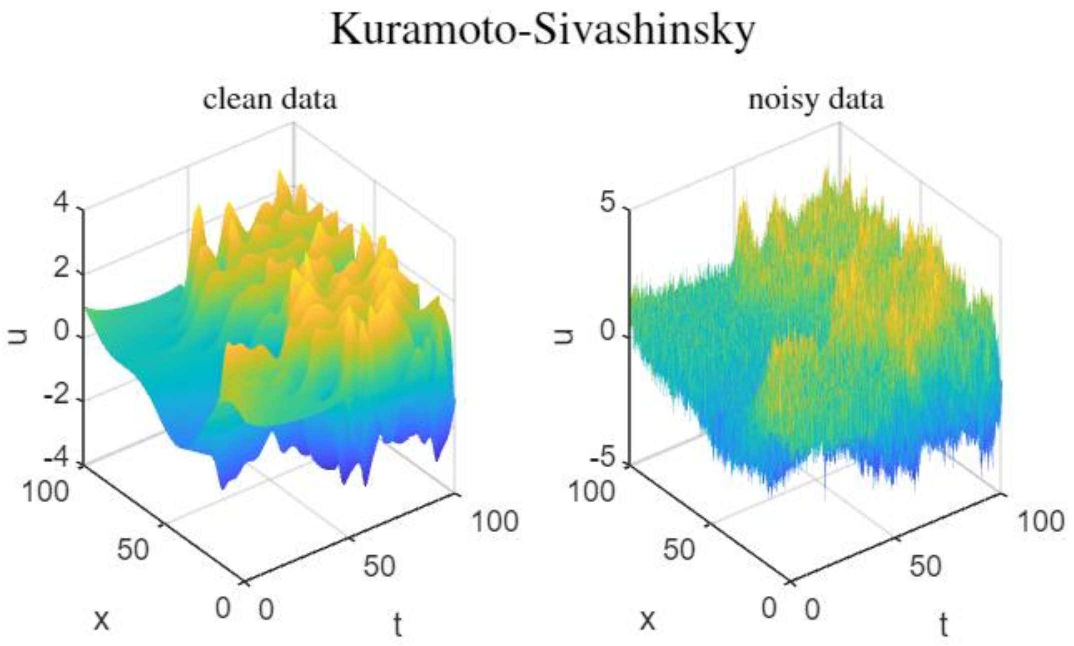
- **Kuramoto-Sivashinsky equation**: $u_t = -uu_x - u_{xx} - u_{xxxx}$
- **Only estimate parameters without selecting models** → $\hat{\xi}_k = \mathrm{argmin}_{\xi_k} \left\| \dot{\mathbf{X}}_k - \mathbf{\Theta}(\mathbf{X})\xi_k \right\|_2^2$



Weak SINDy

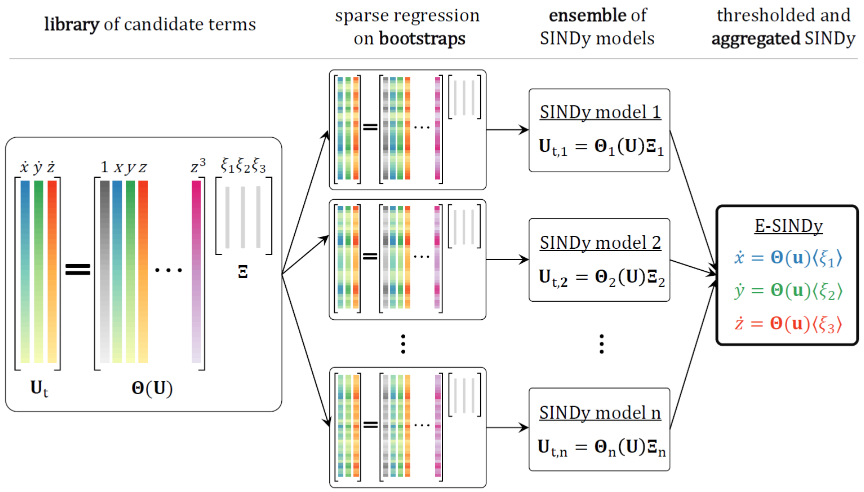
MATLAB example

- **Kuramoto-Sivashinsky equation**: $u_t = -uu_x - u_{xx} - u_{xxxx}$
- **Library**:
  - Advection $u\,u_x$
  - Laplacian u_{xx}
  - Biharmonic u_{xxxx}
  - Linear u
  - 1st order derivative u_x
  - 3rd order derivative u_{xxx}
  - Quadratic uu
  - Cubic uuu

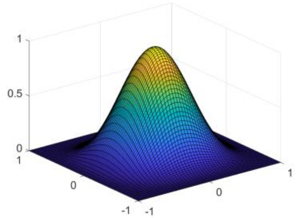


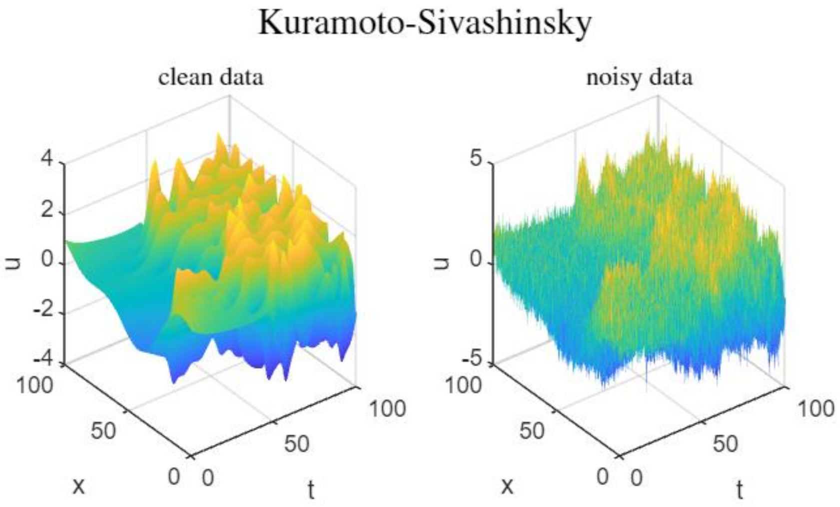
- **Test function**: $\phi(x,t) = (x^2-1)^p(t^2-1)^q$
  - Derivative: $\frac{\partial}{\partial x^d}(x^2-1)^p(t^2-1)^q = \sum_{k=0}^{p}\binom{p}{k}(-1)^k\frac{(2(p-k))!}{(2(p-k)-d)!}x^{2(p-k)-d}(t^2-1)^q$
- **Compare SINDy vs weak-SINDy**
  - Clean and noisy data

Part 4 summary



- **Ensemble SINDy**
  - Ensembling for noise robust model identification
  - Active learning
  - UQ-SINDy





- **Weak-SINDy**
  - Integral and weak form SINDy methods
  - Weak form parameter estimation
  - Weak SINDy KS example

Coding examples

- **MATLAB**
  - Implement different ensemble SINDy methods and test it on the KS data
    - Compare noise robustness, data sampling rate and length
  - Test the weak form SINDy method on different PDEs → Burger's
- **PySINDy**
  - E-SINDy: https://github.com/dynamicslab/pysindy/blob/master/examples/13_ensembling.ipynb
  - Weak SINDy: https://github.com/dynamicslab/pysindy/blob/master/examples/12_weakform_SINDy_examples.ipynb