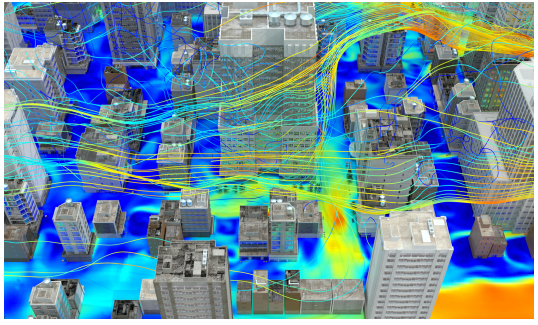
Combatting distribution shift in scientific machine learning

Jiaxi Zhao
joint with Q. Li & S. Arisaka @ NUS, N. Thuerey @ TUM

9th NUS graduate symposium in mathematics
May 13, 2025

Motivation:

Large-scale scientific simulations are unaffordable, e.g. 50M grids, 10s $\Longrightarrow$ 2 days.



Data-driven modeling **may** improve the simulations:

1. Reduced-order model.
2. Surrogate model.

An abstraction of SciML workflow

Simulating the dynamics:

$$\mathcal{L}(\mathbf{u}, \partial_t \mathbf{u}, \mathbf{y}, t) = \mathbf{0}, \quad \mathbf{u} \in \mathcal{U}, \mathbf{y} \in \mathcal{Y}, \mathcal{L} : \mathcal{U} \times T\mathcal{U} \times \mathcal{Y} \times \mathbb{R}_+ \to \mathbb{R}^n,$$
$$\mathbf{y} = \phi(\mathbf{u}, t), \quad \phi : \mathcal{U} \times \mathbb{R}_+ \to \mathbf{y}.$$

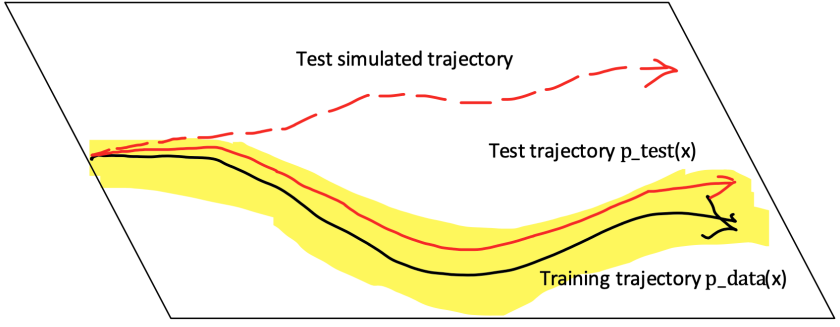
1. $\mathcal{L}$ is known, possibly non-linear, while ϕ is un-known.
2. Datasets: $\{(\mathbf{u}_1, \mathbf{y}_1, t_1), (\mathbf{u}_2, \mathbf{y}_2, t_2), \cdots, (\mathbf{u}_N, \mathbf{y}_N, t_N)\}$.
3. Benchmark algorithm solves the ordinary least square:

$$\arg\min_{\theta} \mathbb{E} \left\| \mathbf{y} - \phi_\theta(\mathbf{u}, t) \right\|^2.$$

Typical examples: subgrid-scale modeling, reynolds stress modeling, exchange-correlation functional.

Dilemma of data-driven scientific computing

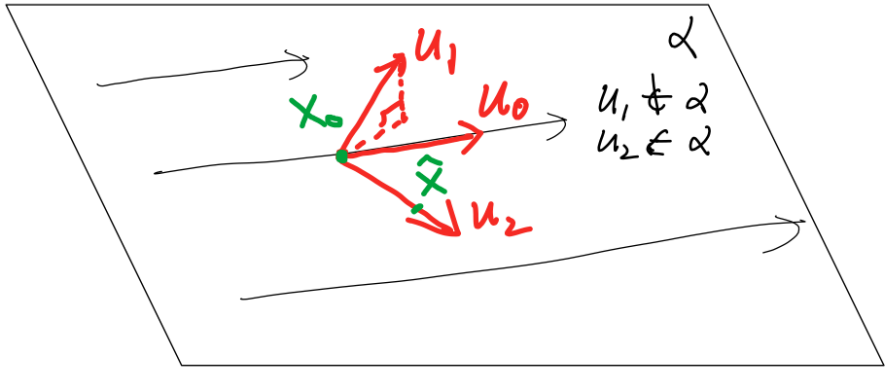
In the data-driven scientific computing, **dynamics structure** can cause **distribution mismatch** between the training and testing data. **The stability of the iterative scheme** is different from classical numerical analysis, e.g. Lax equivalence theorem.



$p\_data(x) \neq p\_test(x)$

Manifold regularization

Regularization encodes the information of the data manifold.



[1]Zhao, Jiaxi, and Qianxiao Li. "Mitigating Distribution Shift in Machine Learning–Augmented Hybrid Simulation." SIAM Journal on Scientific Computing 47.2 (2025): C475-C500.

Linear dynamics

We consider the following the linear hybrid simulation problem

$$\frac{d\mathbf{u}}{dt} = A\mathbf{u} + B\mathbf{y}, \quad \mathbf{u} \in \mathbb{R}^m, \mathbf{y} \in \mathbb{R}^n, A \in \mathbb{R}^{m\times m}, B \in \mathbb{R}^{m\times n}$$
$$\mathbf{y} = C^*\mathbf{u}, \quad C^* \in \mathbb{R}^{n\times m}.$$

Assume the dynamics lie in a low-dimensional subspace V:

$$l_{\mathrm{TR}}(\widehat{C}) := \mathbb{E}_{(\mathbf{u},\mathbf{y})}\left(\left\|\widehat{C}\mathbf{u} - \mathbf{y}\right\|_2^2 + \lambda\left\|P_{V^\perp}(A + B\widehat{C})\mathbf{u}\right\|_2^2\right),$$

For nonlinear case, learn a function $F(\mathbf{u})$ which has data manifold as a level set:

$$l_{\mathrm{TR}}(\theta) := \mathbb{E}_{(\mathbf{u},\mathbf{y})}\left[\|\mathbf{y}_k - \phi_\theta(\mathbf{u})\|_2^2 + \lambda\left((\nabla F(\mathbf{u}))^T L(\mathbf{u}, \phi_\theta(\mathbf{u}), t)\right)^2\right].$$

Design of regularization

The tangent-space regularized estimator has slower error scaling for large λ:

Theorem

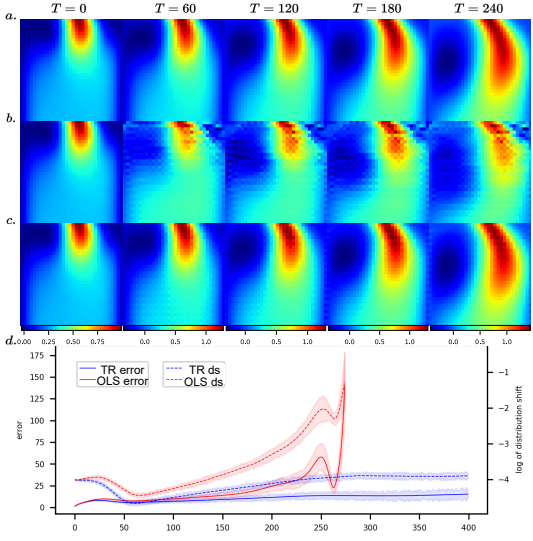
With $Q_m(r, T)$ defined by $m^2 \int_0^T (2 + t^{m-1}) e^{rt} dt$ and

$$e_1 = eig_{\max}(A + B\widehat{C}), \quad e_2 = eig_{\max}((A + B\widehat{C})P_V),$$
$$eig_{\max}(A) = \max\{\Re(s) : \exists v \neq \mathbf{0}, Av = sv\},$$

the errors of OLS and our algorithm are bounded respectively by

$$\mathbb{E}\,\|\widehat{\mathbf{u}}_{OLS}(T) - \mathbf{u}(T)\| \leq c_1\sqrt{\delta}\,\|B\|_2\, Q_m(e_1, T),$$
$$\mathbb{E}\,\|\widehat{\mathbf{u}}_{TR}(T) - \mathbf{u}(T)\| \leq c_2\sqrt{\delta}\Big(\,\|B\|_2\, Q_m(e_2, T)$$
$$+ \frac{9m^4 c_3}{\sqrt{\lambda}}\left(1 + 3m^2 c_1\sqrt{\delta}\,\|B\|_2\right)\left\|A + B\widehat{C}\right\|_2 (1 \vee T^{3m})(1 \vee e^{e_1 T})\Big).$$

Performance



One step further: SGS modeling

$$\begin{aligned}\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j}(u_i u_j) &= -\frac{\partial p}{\partial x_i} + \nu \Delta u_i, \\ \frac{\partial u_i}{\partial x_i} &= 0.\end{aligned}$$

Applying a filter G to the equation, i.e. $\overline{u} = G * u$

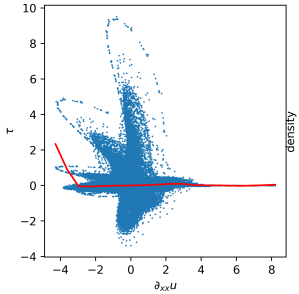
$$\begin{aligned}\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u}_i \overline{u}_j) &= -\frac{\partial \overline{p}}{\partial x_i} + \nu \Delta \overline{u}_i - \frac{\partial \tau_{ij}}{\partial x_j}, \\ \frac{\partial \overline{u}_i}{\partial x_i} &= 0, \\ \tau_{ij} &= \overline{u_i u_j} - \overline{u}_i \overline{u}_j.\end{aligned}$$

$\min_\theta \mathbb{E} \left\| \tau - \phi_\theta(\overline{\mathbf{u}}) \right\|^2 \Longrightarrow$ Blow up simulation or wrong statistics.

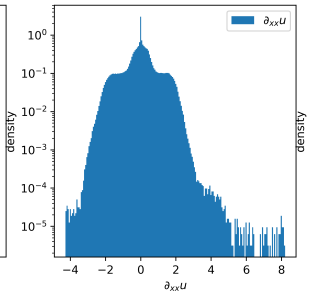
Understanding the SGS modeling

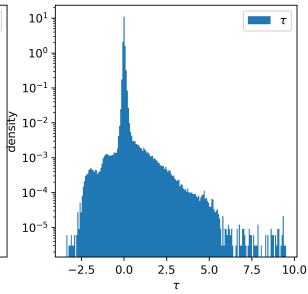
Kuramoto–Sivashinsky equation:

$$u_t = -(c+u)u_x - uu_x - u_{xx} - \nu u_{xxxx}, \quad u(0,t) = u(L,t) = u_x(0,t) = u_x(L,t) = 0, \forall t.$$



Multivaluedness





Data imbalance

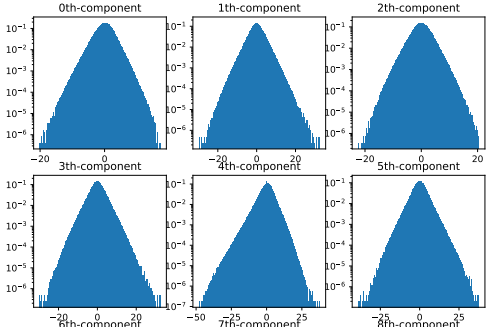
Understanding the SGS modeling

1. Boundary layer (BL) and multiscale structure causes the data imbalance. For wall-bounded NS equation:

   BL of thickness $\delta \propto \sqrt{\nu} \Longrightarrow O(\nu^{-1/2})$ gradients for $O(\sqrt{\nu})$ portion of data. Bulk fluid has $O(1)$ gradients.

2. Dimension reduction caused the multivaluedness: Mori-Zwanzig formalism.

Sanity check on homogeneous isotropic turbulence (HIT) data:



Solution: Probabilistic ansatz

Regression to generative modeling:

$$\tau = \phi_\theta(\mathbf{u}) \quad \rightarrow \quad \tau \sim p_\theta(\cdot|\mathbf{u}).$$

Change of the loss functions:

$$\min_\theta \sum_n \left\| \phi_\theta(\widetilde{\mathbf{u}}^{(n)}) - \tau^{(n)} \right\|^2, \quad \max_\theta \sum_{i=n}^{N} \log p_\theta(\tau^{(n)}|\mathbf{u}^{(n)}).$$

[2] Zhao, Jiaxi, Sohei Arisaka, and Qianxiao Li. "Generative subgrid-scale modeling." ICLR 2025 Workshop on Machine Learning Multiscale Processes.

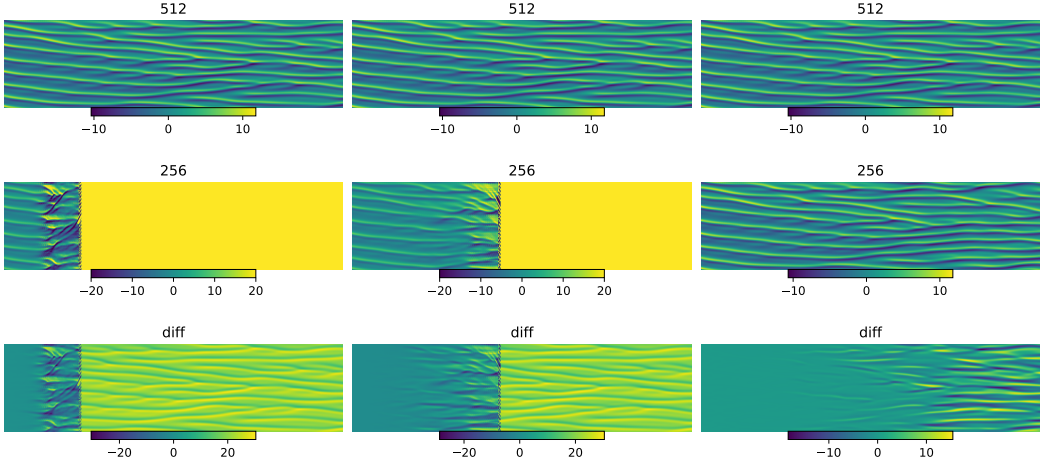
Integrating in the simulation

How can we deploy this model in the simulation?

$$\begin{aligned}
\text{Gaussian: } \mathbf{u}_i &\Longrightarrow \mu_\theta(\mathbf{u}_i), \sigma_\theta(\mathbf{u}_i), z \sim N(0,1), \\
\tau_{ij} &= \mu_\theta(\mathbf{u}_i) + \sigma_\theta(\mathbf{u}_i) z, \\
\text{Gaussian mixture: } \mathbf{u}_i &\Longrightarrow \mu_\theta^j(\mathbf{u}_i), \sigma_\theta^j(\mathbf{u}_i), z \sim N(0,1), j \sim [M], \\
\tau_i &= \mu_\theta^j(\mathbf{u}_i) + \sigma_\theta^j(\mathbf{u}_i) z,
\end{aligned}$$

There is temporal and spatial consistency issue. Should we use the same latent variable z for all the grid points at all the time step or we should use different z_i for different $\mathbf{u}_i$?

Comparison with NN ansatz and Smagorinsky model ansatz



(a) Regression ansatz: neural network

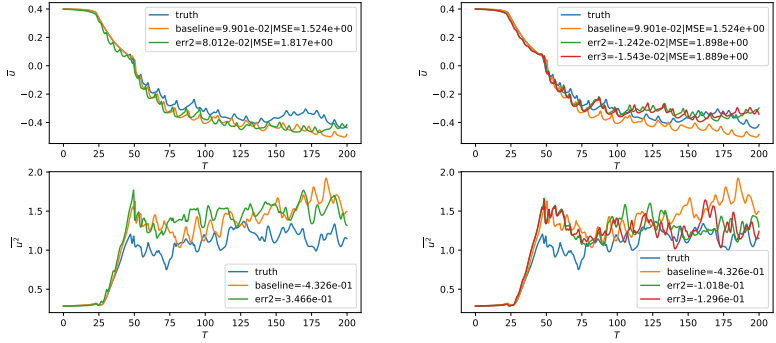
(b) Regression ansatz: Smagorinsky

(c) Probabilistic ansatz: Gaussian

Comparison with the regression-based method

$$\langle \overline{u} \rangle = \frac{1}{LT} \int_{[0,L]} \int_{t}^{t+T} u(x,t) dt dx,$$

$$\left\langle \overline{u^2} \right\rangle = \frac{1}{LT} \int_{[0,L]} \int_{t}^{t+T} u^2(x,t) dt dx,$$



Code development

`https://github.com/jiaxi98/ml4dynamics`
A codebase for testing various algorithms for data-driven hybrid simulations,

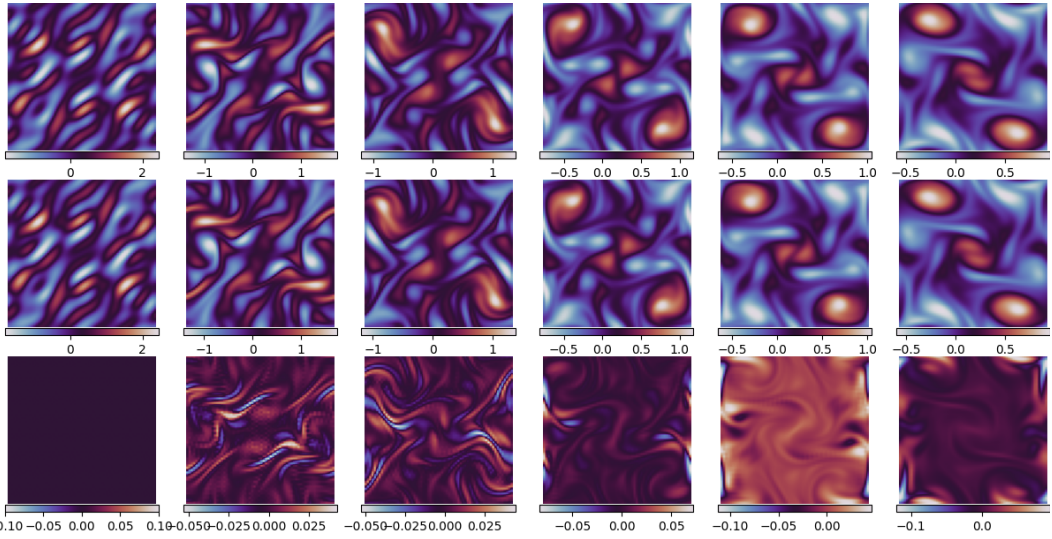
1. Regularization-based method.
2. Generative modeling.
3. Active-learning method.

`https://github.com/jiaxi98/pyfoam`
Data-driven turbulence modeling platform based on OpenFOAM.

Backup slides

	baseline	regression	gaussian, fix	gaussian sample
A priori error	NA	0.976	-2.173	-2.173
$\int \|\mathbf{u} - \mathbf{u}_0\|_2^2 \, dxdt$	1.524	2.036	1.720	1.597
$\langle \overline{u} \rangle - \langle \overline{u_0} \rangle$	9.901E-02	1.011E-01	3.870E-02	3.214E-02
$\left\langle \overline{u^2} \right\rangle - \left\langle \overline{u_0^2} \right\rangle$	-4.326E-01	-4.241E-01	-1.895E-01	-6.577E-02
A priori error	NA	0.987	-2.583	-2.583
$\int \|\mathbf{u} - \mathbf{u}_0\|_2^2 \, dxdt$	1.524	1.817	1.898	1.889
$\langle \overline{u} \rangle - \langle \overline{u_0} \rangle$	9.901E-02	8.012E-02	-1.242E-02	-1.543E-02
$\left\langle \overline{u^2} \right\rangle - \left\langle \overline{u_0^2} \right\rangle$	-4.326E-01	-3.466E-01	-1.018E-01	-1.296E-01

Backup slides



Backup slides