Modeling stochastic systems using physics-informed deep generative models

Paris Perdikaris & Yibo Yang
Department of Mechanical Engineering and Applied Mechanics
University of Pennsylvania
email: pgp@seas.upenn.edu

New York Scientific Data Summit
Columbia University
June 14th, 2019

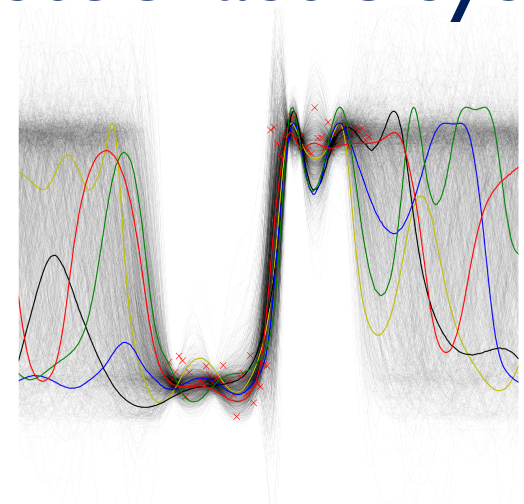
UNIVERSITY of PENNSYLVANIA

Data-driven modeling of stochastic systems

$\boldsymbol{z} \sim p(\boldsymbol{z}) \longrightarrow$

$\boldsymbol{x}, t \longrightarrow$

$\boldsymbol{u}_t + \mathcal{N}_{\boldsymbol{x}} \boldsymbol{u} = 0 \longrightarrow$



$p(\boldsymbol{u}|\boldsymbol{x}, t)$

Physics-informed deep generative models: $p(\boldsymbol{u}|\boldsymbol{x}, t, \boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z}), \quad \text{such that} \quad \boldsymbol{u}_t + \mathcal{N}_{\boldsymbol{x}} \boldsymbol{u} = 0.$

Current approaches (non-intrusive methods):

- Polynomial chaos, sparse grid quadratures, multi-level/multi-fidelity Monte Carlo, reduced order/surrogate models (POD, Gaussian processes, etc.)
- All face limitations in modeling high-dimensional stochastic systems.
- All require repeated evaluations of expensive simulators/experiments.

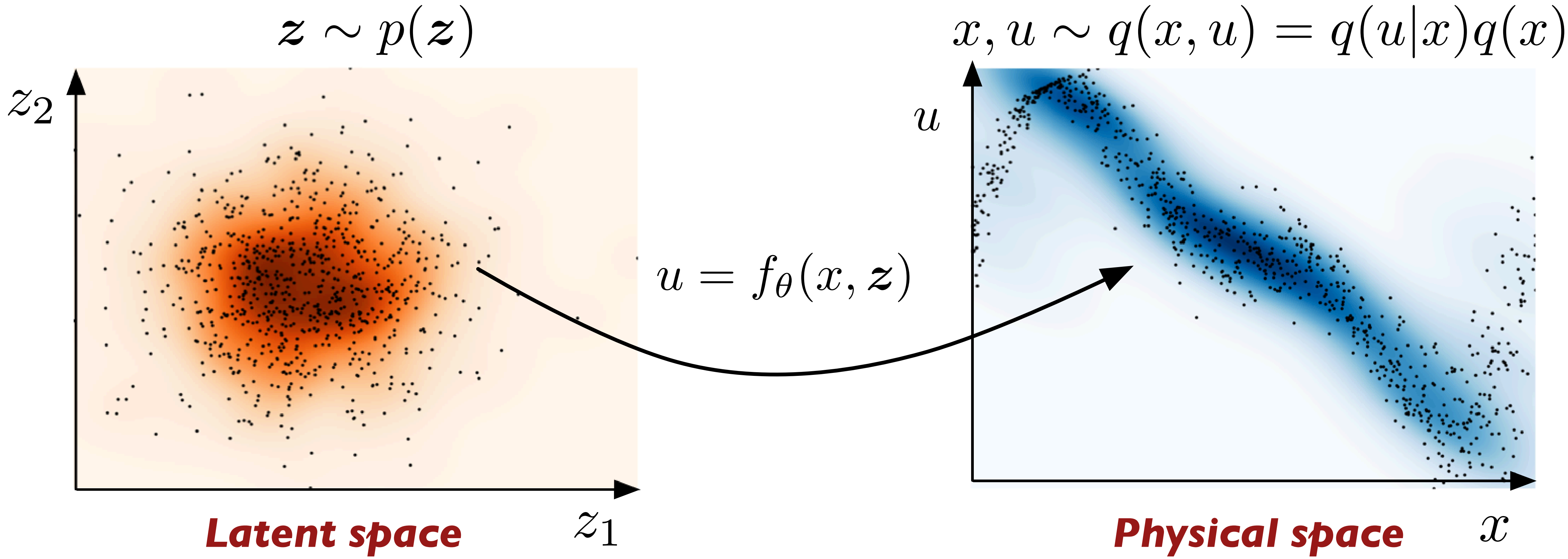
Goal of this work:

- Introduce a probabilistic deep learning framework for modeling stochastic systems that entirely bypasses the need for repeatedly sampling expensive experiments or numerical simulators.

Approach:

- Build deep generative models (GANs, VAEs, etc.) with physics-informed constraints.
- Develop robust statistical inference algorithms for approximating complex conditional distributions directly from noisy data.

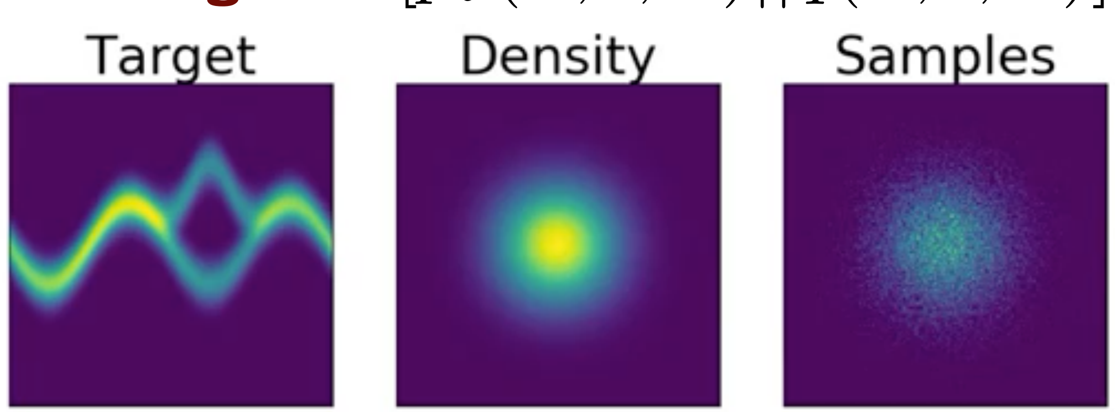
Conditional deep generative models



Model: $\boldsymbol{u} = f_\theta(\boldsymbol{x}, t, \boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z}), \quad \text{such that} \quad \boldsymbol{u}_t + \mathcal{N}_{\boldsymbol{x}}\boldsymbol{u} = 0$

$$p_\theta(\boldsymbol{u}|\boldsymbol{x}, t) = \int p_\theta(\boldsymbol{u}, \boldsymbol{z}|\boldsymbol{x}, t)d\boldsymbol{z} = \int p_\theta(\boldsymbol{u}|\boldsymbol{x}, t, \boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}$$

Training: $\mathbb{KL}[p_\theta(\boldsymbol{x}, t, \boldsymbol{u})||q(\boldsymbol{x}, t, \boldsymbol{u})]$



Key ingredients:

1. Density ratio estimation via probabilistic classification.
2. Joint distribution matching via adversarial inference with entropy regularization.
3. Physics-informed constraints for generating samples that approximately satisfy the underlying PDE.

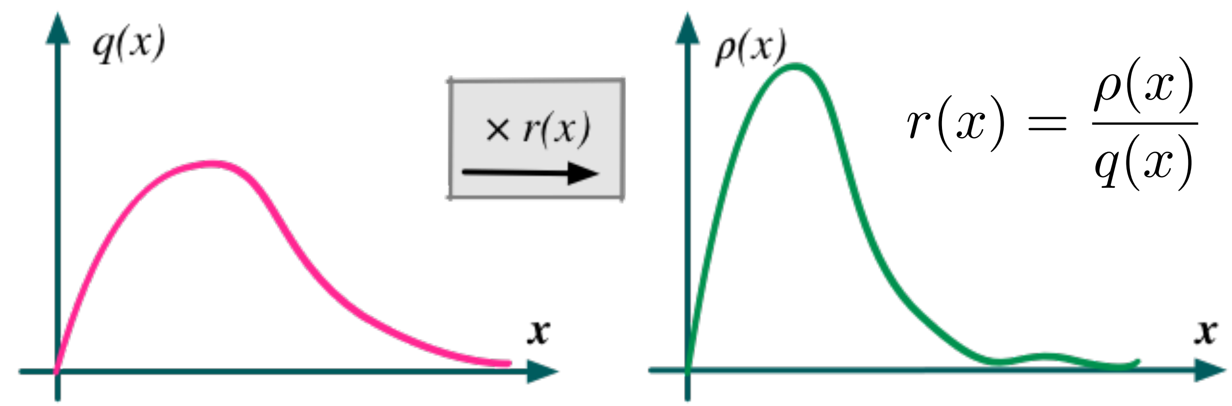
#1: Density ratio estimation via probabilistic classification

$$\mathbb{KL}[p_\theta(\boldsymbol{x}, t, \boldsymbol{u}) || q(\boldsymbol{x}, t, \boldsymbol{u})] := \mathbb{E}_{p_\theta(\boldsymbol{x}, t, \boldsymbol{u})} \left[\log \left(\frac{p_\theta(\boldsymbol{x}, t, \boldsymbol{u})}{q(\boldsymbol{x}, t, \boldsymbol{u})} \right) \right]$$

Estimating density ratios is a challenging task:

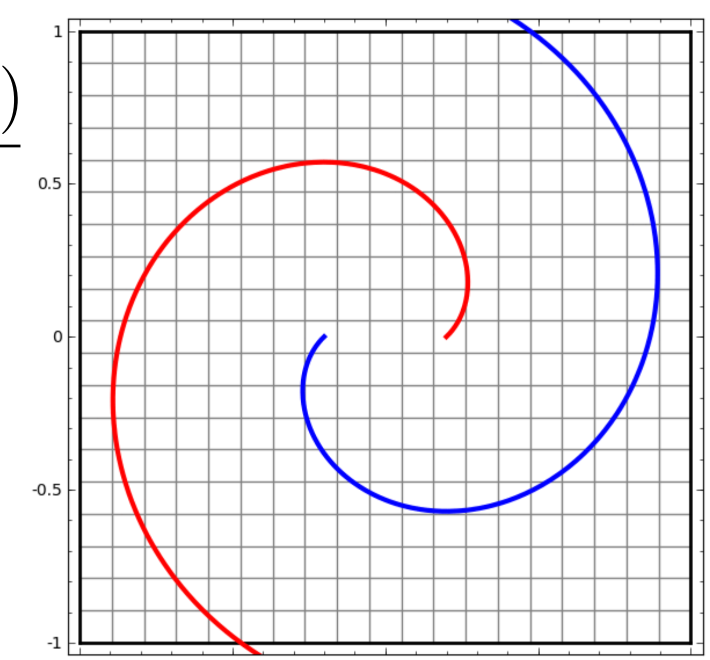
- Each part of the ratio may itself involve intractable integrals
- We often deal with high-dimensional quantities.
- We may only have samples drawn from the two distributions, not their analytical forms.

This is where the ***density ratio trick*** enters: it allows us to construct a binary classifier that distinguishes between samples from the two distributions.



The density ratio gives the correction factor needed to make two distributions equal.

$$\begin{aligned}
\frac{p_\theta(\boldsymbol{x}, t, \boldsymbol{u})}{q(\boldsymbol{x}, t, \boldsymbol{u})} &= \frac{\rho(\boldsymbol{x}, t, \boldsymbol{u} | y = +1)}{\rho(\boldsymbol{x}, t, \boldsymbol{u} | y = -1)} \\
&= \frac{\rho(y = +1 | \boldsymbol{x}, t, \boldsymbol{u}) \rho(\boldsymbol{x}, t, \boldsymbol{u})}{\rho(y = +1)} \Big/ \frac{\rho(y = -1 | \boldsymbol{x}, t, \boldsymbol{u}) \rho(\boldsymbol{x}, t, \boldsymbol{u})}{\rho(y = -1)} \\
&= \frac{\rho(y = +1 | \boldsymbol{x}, t, \boldsymbol{u})}{\rho(y = -1 | \boldsymbol{x}, t, \boldsymbol{u})} = \frac{\rho(y = +1 | \boldsymbol{x}, t, \boldsymbol{u})}{1 - \rho(y = +1 | \boldsymbol{x}, t, \boldsymbol{u})} \\
&= \frac{T(\boldsymbol{x}, t, \boldsymbol{u})}{1 - T(\boldsymbol{x}, t, \boldsymbol{u})}.
\end{aligned}$$

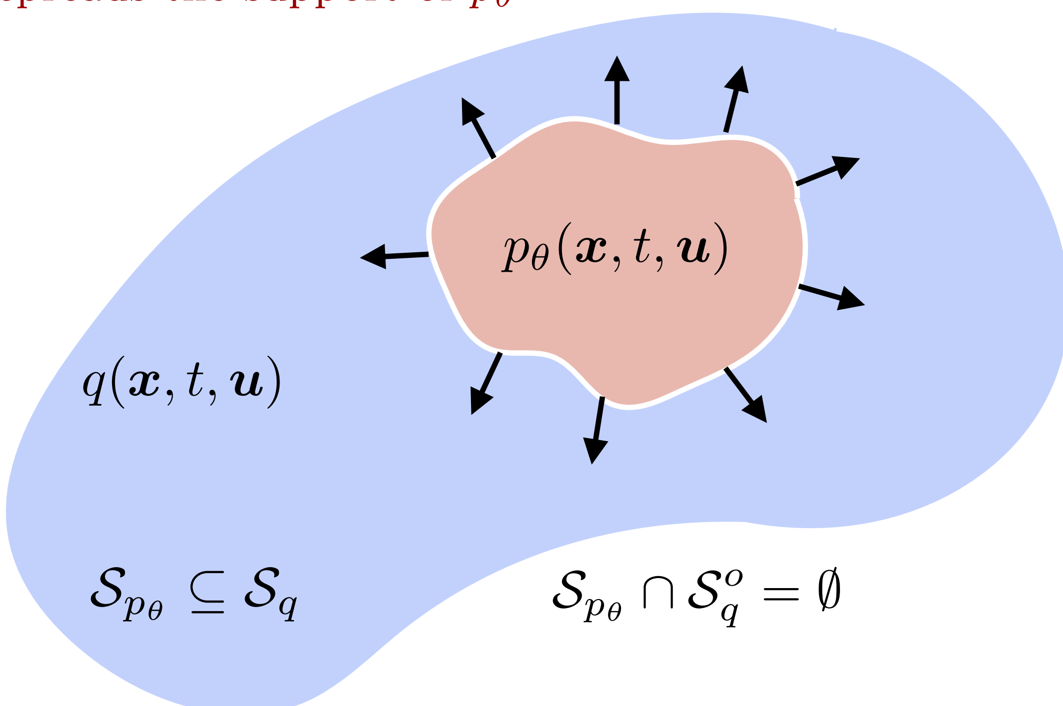


#2: Joint distribution matching

- Physics-informed generative model : $p(\boldsymbol{u}|\boldsymbol{x}, t, \boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z}),$ such that $\boldsymbol{u}_t + \mathcal{N}_{\boldsymbol{x}}\boldsymbol{u} = 0.$
- We train the model via joint distribution matching by minimizing the reverse $\mathbb{KL}$-divergence :

$$\mathbb{KL}[p_\theta(\boldsymbol{x}, t, \boldsymbol{u})||q(\boldsymbol{x}, t, \boldsymbol{u})] = -h(p_\theta(\boldsymbol{x}, t, \boldsymbol{u}))) - \mathbb{E}_{p_\theta(\boldsymbol{x},t,\boldsymbol{u})}[\log(q(\boldsymbol{x}, t, \boldsymbol{u}))]$$

$$= \underbrace{-h(p_\theta(\boldsymbol{x}, t, \boldsymbol{u})))}_{\text{spreads the support of } p_\theta} - \int_{\mathcal{S}_{p_\theta} \cap \mathcal{S}_q} \log(q(\boldsymbol{x}, t, \boldsymbol{u}))p_\theta(\boldsymbol{x}, t, \boldsymbol{u})d\boldsymbol{x}dtd\boldsymbol{u} \underbrace{- \int_{\mathcal{S}_{p_\theta} \cap \mathcal{S}_q^o} \log(q(\boldsymbol{x}, t, \boldsymbol{u}))p_\theta(\boldsymbol{x}, t, \boldsymbol{u})d\boldsymbol{x}dtd\boldsymbol{u}}_{\text{penalizes non-overlaps of } p_\theta \text{ and } q}$$



$p_\theta(\boldsymbol{x}, t, \boldsymbol{u})$: Generative model distribution

$q(\boldsymbol{x}, t, \boldsymbol{u})$: Empirical data distribution

- Variational bound for the intractable entropy term :

$$\begin{aligned} h(p_\theta(\boldsymbol{x}, t, \boldsymbol{u})) &= h(p(\boldsymbol{z})) - h(p_\theta(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u})) + \cancelto{0}{h(p_\theta(\boldsymbol{x}, t, \boldsymbol{u}|\boldsymbol{z}))} \\ &= h(p(\boldsymbol{z})) + \mathbb{E}_{p_\theta(\boldsymbol{x},t,\boldsymbol{u},\boldsymbol{z})}[\log(p_\theta(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u}))] \\ &= h(p(\boldsymbol{z})) + \mathbb{E}_{p_\theta(\boldsymbol{x},t,\boldsymbol{u},\boldsymbol{z})}[\log(q_\phi(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u}))] \\ &\quad + \mathbb{E}_{p_\theta(\boldsymbol{x},t,\boldsymbol{u})}[\mathbb{KL}[p_\theta(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u})||q_\phi(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u})]] \\ &\geq h(p(\boldsymbol{z})) + \mathbb{E}_{p_\theta(\boldsymbol{x},t,\boldsymbol{u},\boldsymbol{z})}[\log(q_\phi(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u}))]. \end{aligned}$$

Remarks:

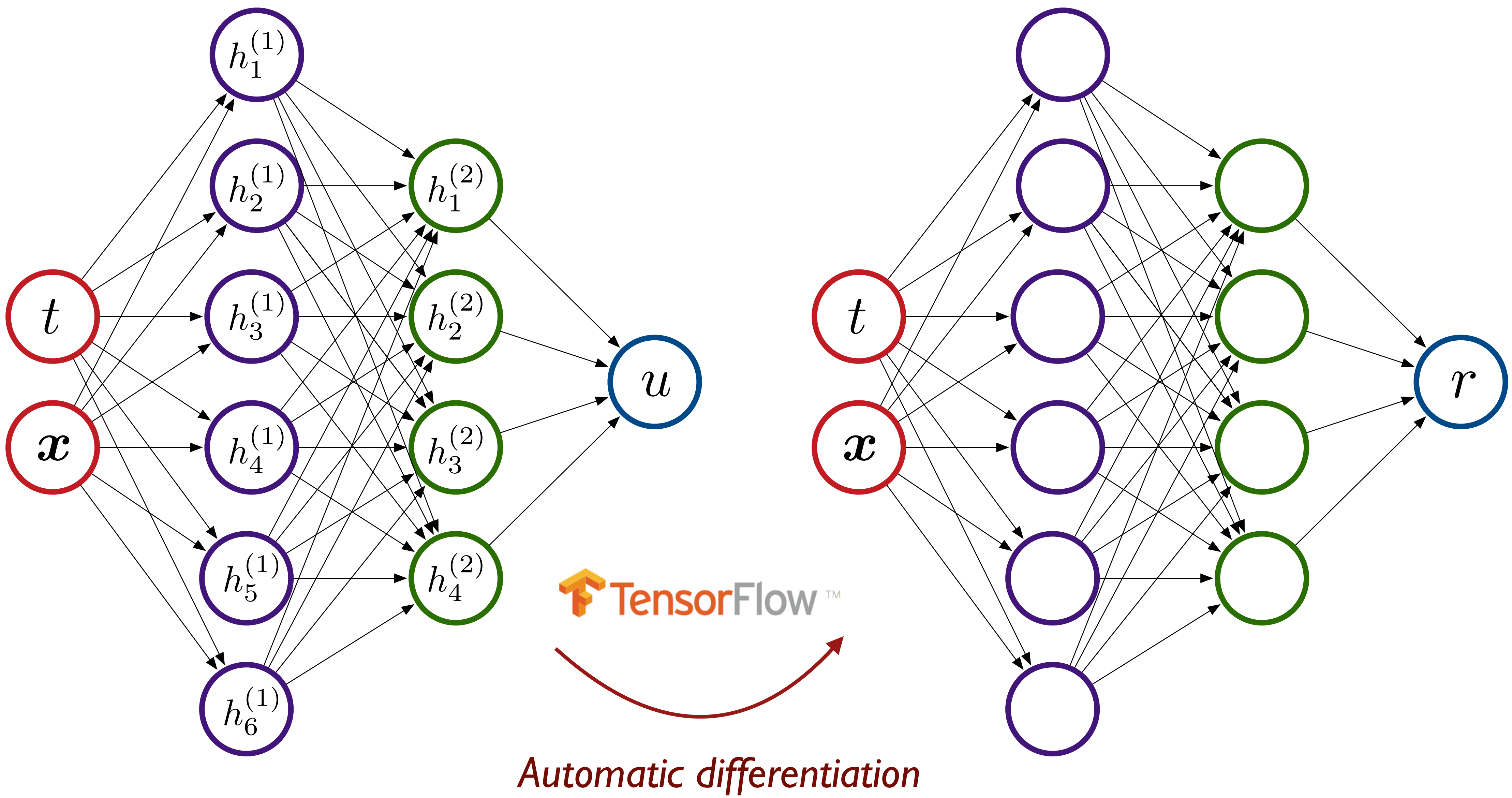
- Flexible variational inference with implicit distributions, no mean field approximations.
- Explicit control over the pathology of mode-collapse in GANs.
- Approximate posterior inference over latent variables (not possible with GANs).
- Discovery of disentangled representations via cycle-consistency in latent space.

#3: Physics-informed constraints

$f_\theta(\boldsymbol{x}, t)$: Neural network

$r_\theta(\boldsymbol{x}, t)$: Physics-informed neural network

$$\begin{cases} [\boldsymbol{x}, t] \xrightarrow{f_\theta} \boldsymbol{u}(\boldsymbol{x}, t) \\ [\boldsymbol{x}, t] \xrightarrow{r_\theta} \frac{\partial}{\partial t} f_\theta(\boldsymbol{x}, t) + \mathcal{N}_{\boldsymbol{x}} f_\theta(\boldsymbol{x}, t) \end{cases}$$



$$\mathcal{L}_{\mathrm{PDE}}(\theta) := \underbrace{\frac{1}{N_r} \sum_{I=1}^{N_r} \| r_\theta(\boldsymbol{x}, t) - \boldsymbol{r}_i \|^2}_{\text{physics-informed regularization}}$$

Lagaris, I. E., Likas, A., & Fotiadis, D. I. (1998). Artificial neural networks for solving ordinary and partial differential equations. IEEE transactions on neural networks, 9(5), 987-1000.

Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378, 686-707.

Adversarial inference for physics-informed deep generative models

- Adversarial optimization:

$$\max_{\psi} \ \mathcal{L}_{\mathcal{D}}(\psi)$$

$$\min_{\theta,\phi} \ \mathcal{L}_{\mathcal{G}}(\theta,\phi) + \beta \mathcal{L}_{PDE}(\theta),$$

$p(\boldsymbol{z}) \longrightarrow$ ***prior***

$f_\theta(\boldsymbol{x}, t, \boldsymbol{z}) \longrightarrow$ ***generator***

$T_\psi(\boldsymbol{x}, t, \boldsymbol{u}) \longrightarrow$ ***discriminator***

$q_\phi(\boldsymbol{z}|\boldsymbol{x}, t, \boldsymbol{u}) \longrightarrow$ ***encoder***

- Generator and discriminator loss functions:

$$\mathcal{L}_{\mathcal{D}}(\psi) = \mathbb{E}_{q(\boldsymbol{x},t)p(\boldsymbol{z})}[\log \sigma(T_\psi(\boldsymbol{x}, t, f_\theta(\boldsymbol{x}, t, \boldsymbol{z})))] + \mathbb{E}_{q(\boldsymbol{x},t,\boldsymbol{u})}[\log(1 - \sigma(T_\psi(\boldsymbol{x}, t, \boldsymbol{u})))]$$

$$\mathcal{L}_{\mathcal{G}}(\theta,\phi) = \mathbb{E}_{q(\boldsymbol{x},t,\boldsymbol{u})p(\boldsymbol{z})}[T_\psi(\boldsymbol{x}, t, f_\theta(\boldsymbol{x}, t, \boldsymbol{z})) + (1-\lambda)\log(q_\phi(\boldsymbol{z}|\boldsymbol{x}, t, f_\theta(\boldsymbol{x}, t, \boldsymbol{z})))]$$

- PDE constraints:

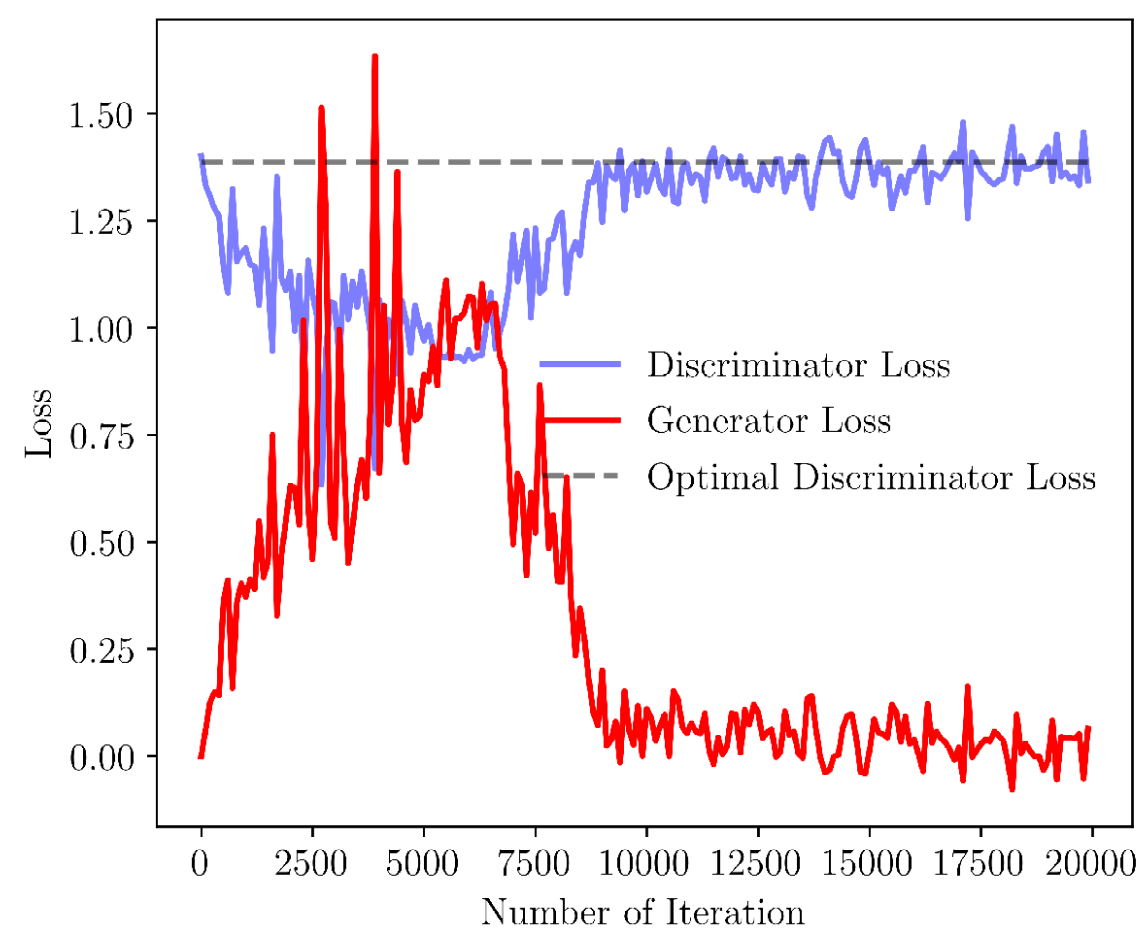
$$\mathcal{L}_{\text{PDE}}(\theta) := \underbrace{\frac{1}{N_r}\sum_{I=1}^{N_r} \|r_\theta(\boldsymbol{x}, t) - \boldsymbol{r}_i\|^2}_{\text{physics-informed regularization}}$$

- Alternate stochastic gradient updates between the generator and the discriminator.

- Optimal discriminator loss:

$$\ln(4) = -2 \times \ln(0.5) = 1.384$$

Stochastic gradient descent dynamics



Yang, Y., & Perdikaris, P. (2019). Adversarial uncertainty quantification in physics-informed neural networks. Journal of Computational Physics.

A pedagogical example

Problem setup:

$$u_{xx} - u^2 u_x = f(x), \qquad x \in [-1, 1],$$

$$f(x) = -\pi^2 \sin(\pi x) - \pi \cos(\pi x) \sin^2(\pi x)$$

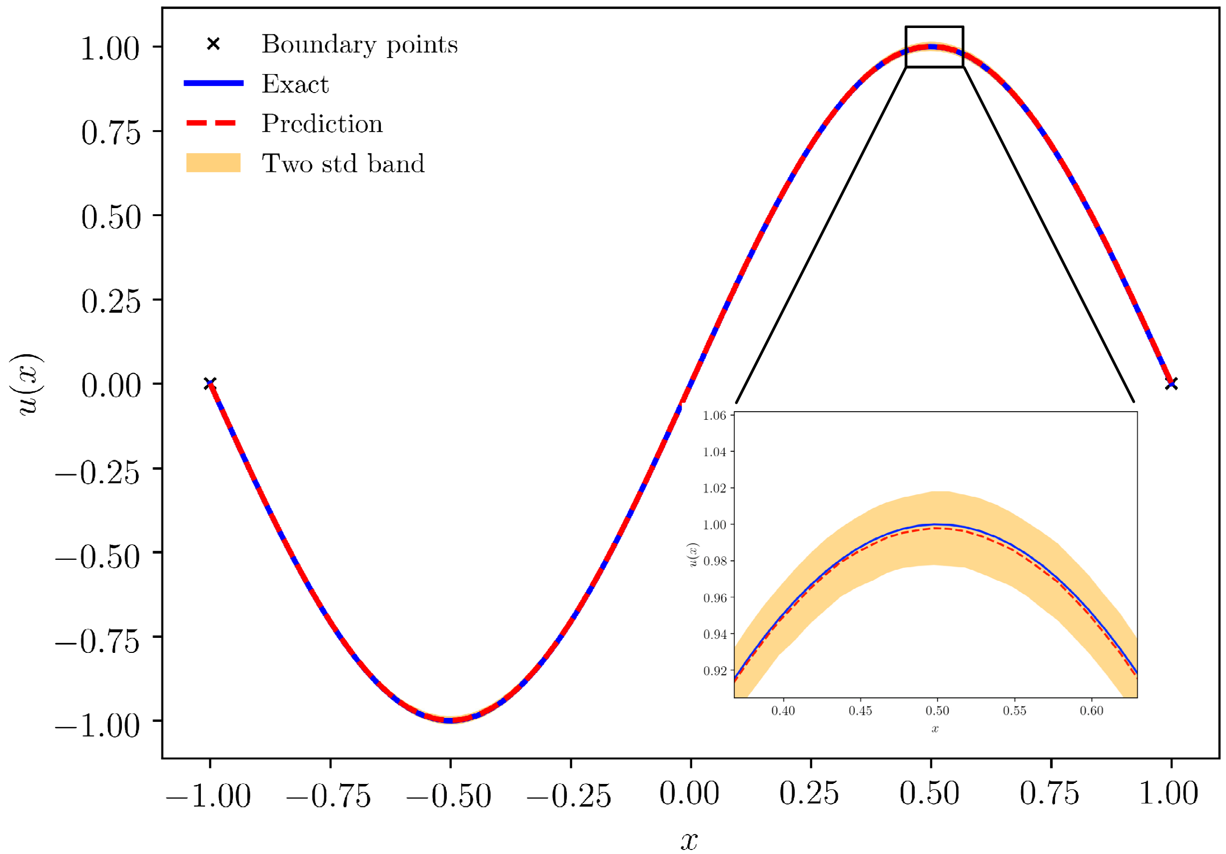
$$u(-1), u(1) \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \boldsymbol{I})$$

Model setup and training data:

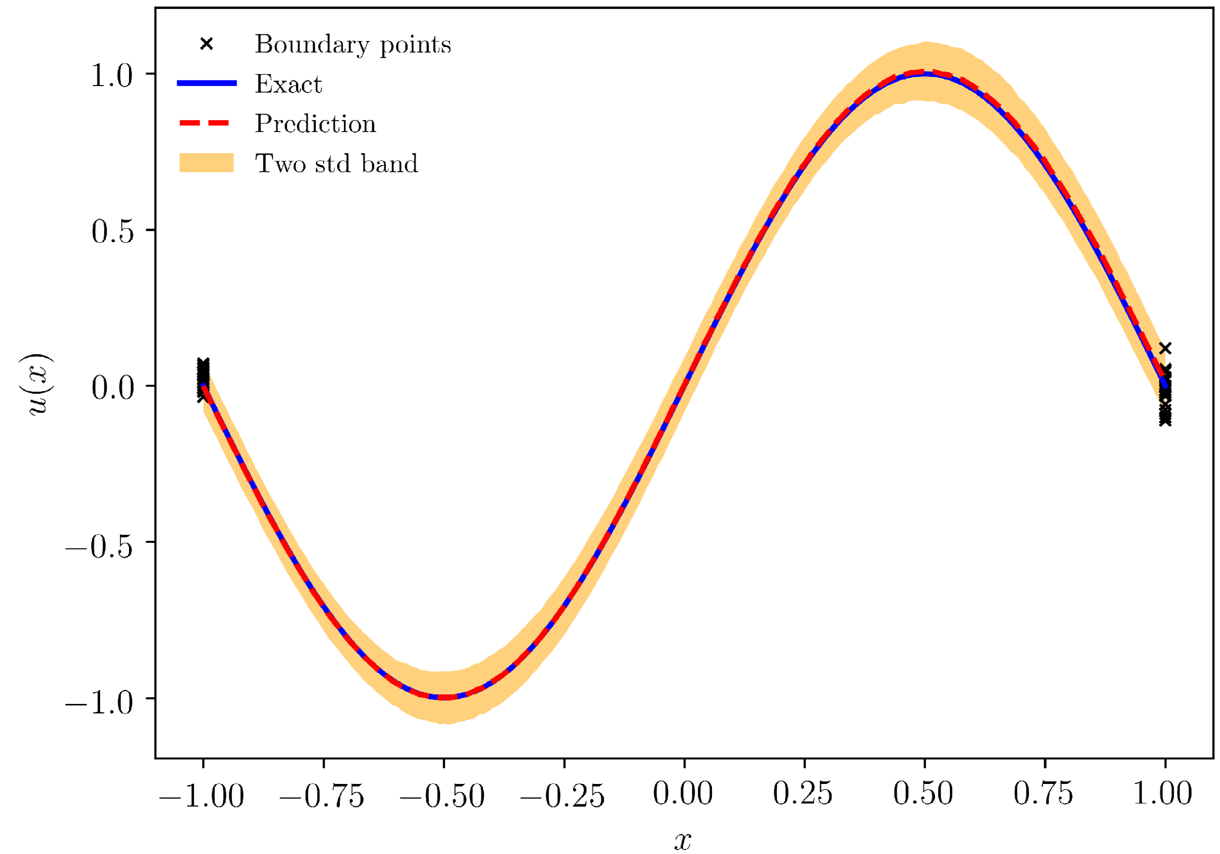
- 20 realizations at each boundary point
- 100 collocation points for enforcing the PDE residual
- $\lambda = 1.5, \beta = 1.0$

Neural nets: Feed-forward with 2 hidden layers, 50 neurons, tanh() activation, Adam optimizer.

$\sigma_n^2 = 0.0$ *(deterministic case)*



$\sigma_n^2 = 0.05$ *(stochastic case)*



Mean and two standard deviations of $p_\theta(u|x, z)$

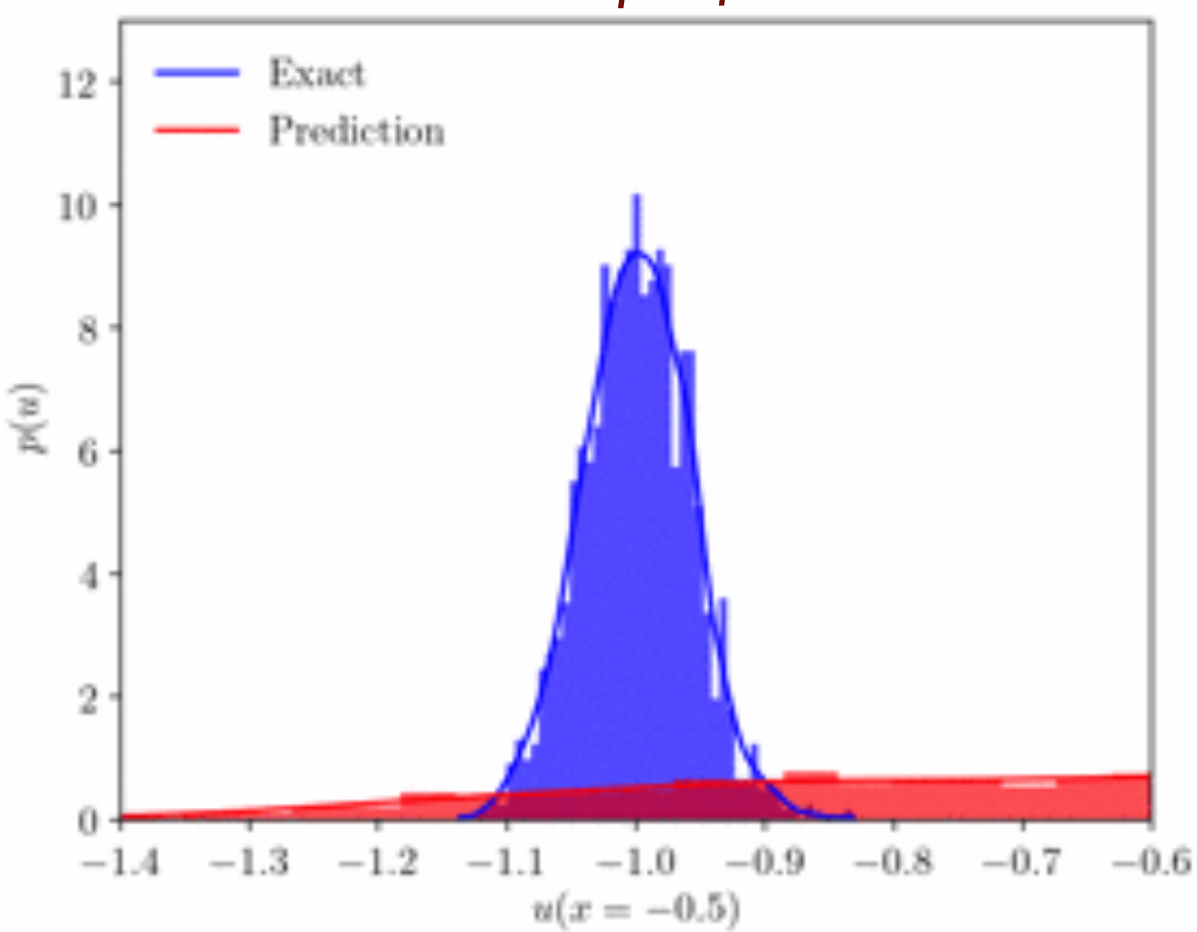
Yang, Y., & Perdikaris, P. (2019). Adversarial uncertainty quantification in physics-informed neural networks. Journal of Computational Physics.

A pedagogical example

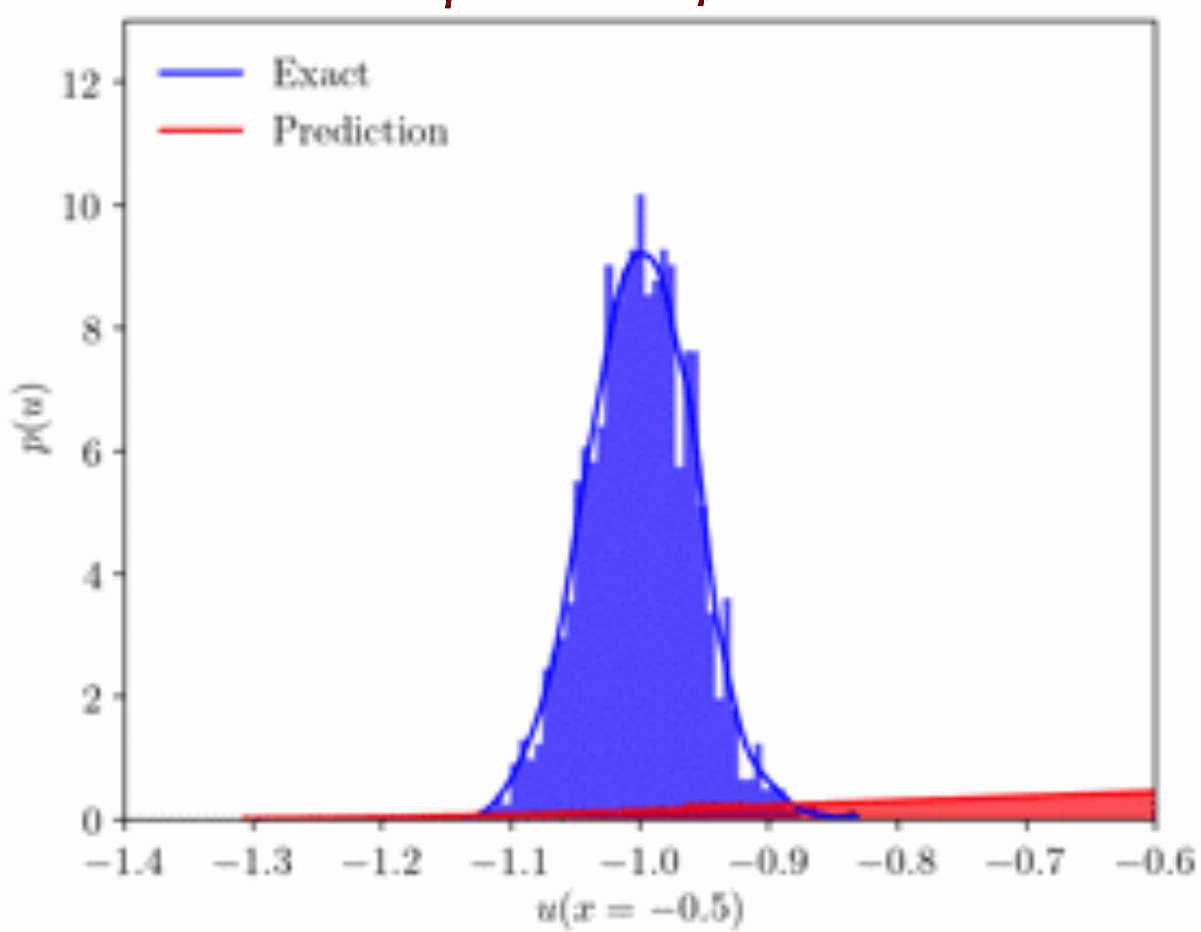
Sensitivity wrt λ, β $\mathbb{E}_{p(x)}\{\mathbb{KL}[p_\theta(u|x)||q(u|x)]\}$

β \ λ	1.0	1.5	2.0	5.0
0	5.0e+05	7.5e+01	6.0e+01	4.4e+01
1.0	3.3e+02	1.8e-01	2.9e-01	2.0e-01
2.0	2.1e+02	1.7e-01	5.0e-02	1.2e-01
5.0	3.5e+01	1.8e-01	1.9e-01	1.1e-01

Mode-collapse for $\lambda = 1.0$



Stable prediction for $\lambda = 1.5$



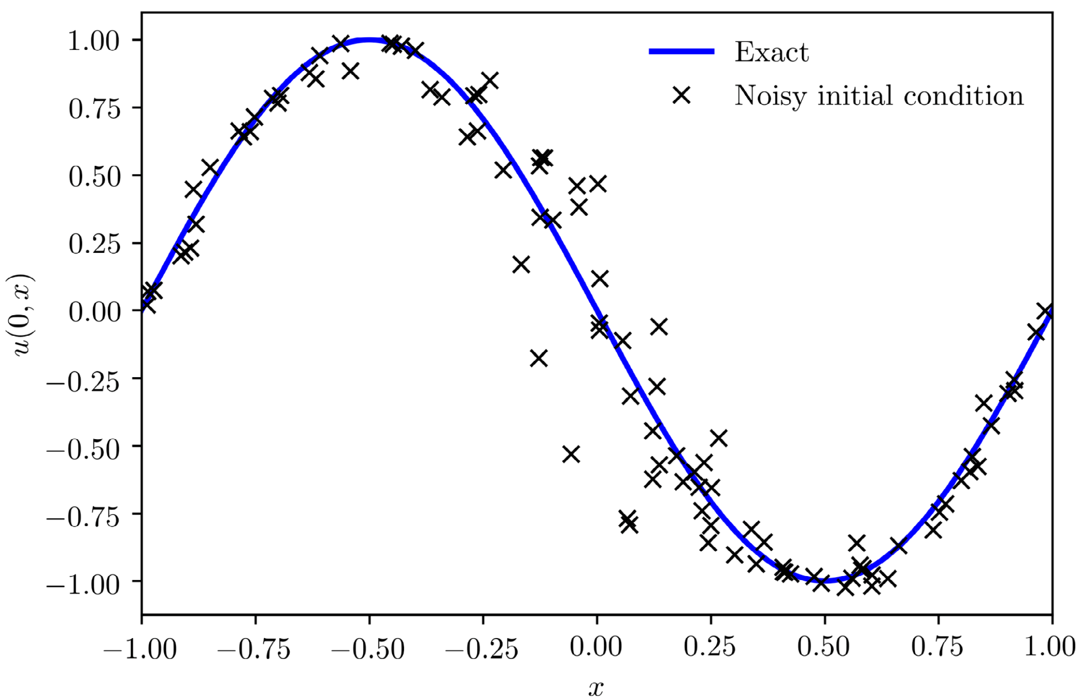
Uncertainty propagation in nonlinear conservation laws

Problem setup:
Burgers equation

$$\begin{cases} u_t + uu_x - \nu u_{xx} = 0, \\ u(0,x) = -\sin(\pi x), \\ u(t,-1) = u(t,1) = 0, \\ x \in [-1,1], t \in [0,1], \\ \nu = 0.01/\pi \end{cases}$$

Model setup and training data:

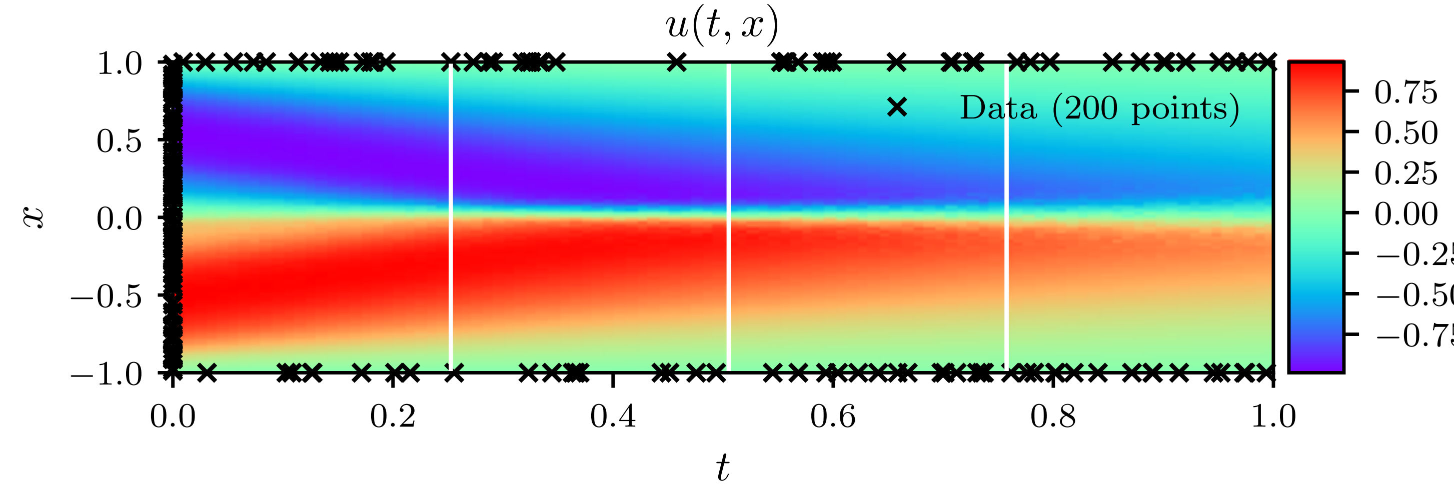
- 100 scattered measurements from a random initial condition
- 10,000 collocation points for enforcing the PDE residual
- $\lambda = 1.5, \beta = 1.0$

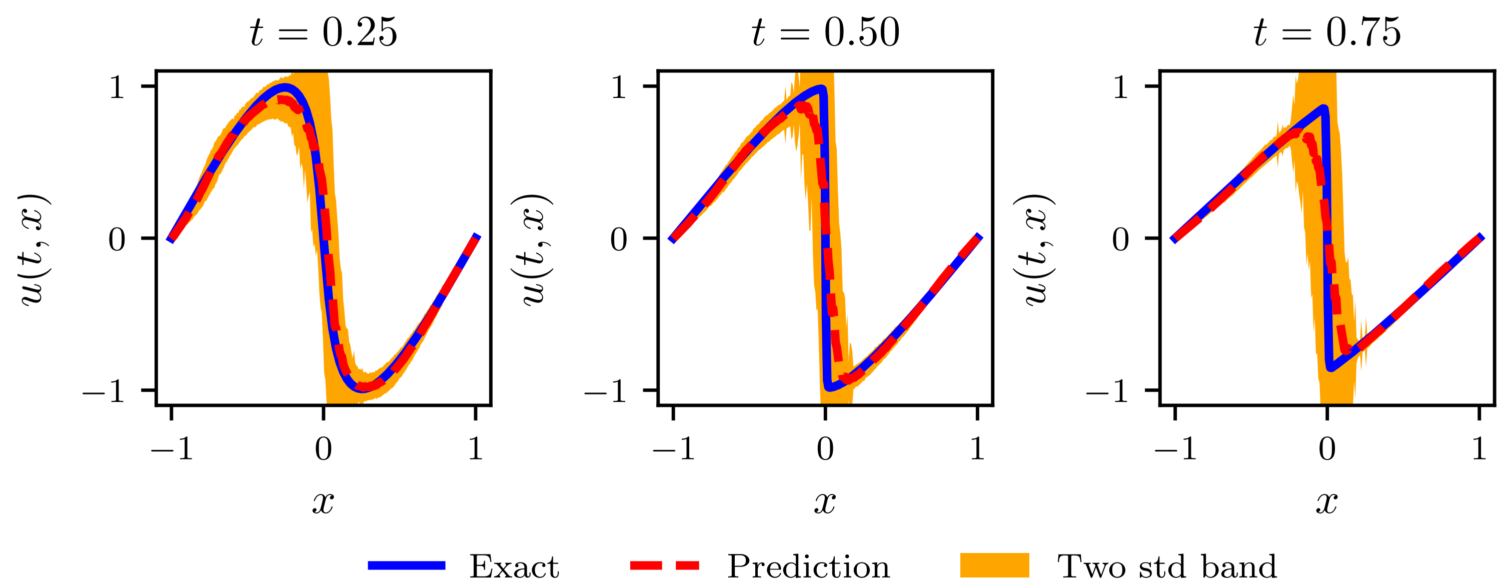


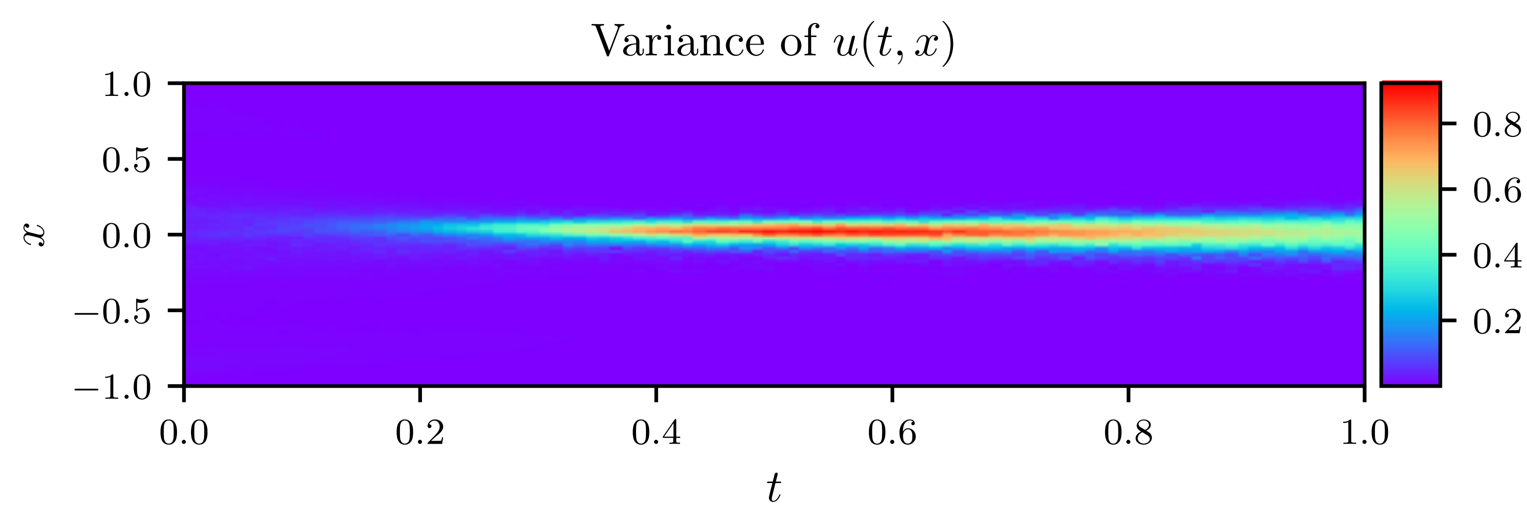
Random initial condition:

$$u(x,0) = -\sin(\pi(x+2\delta)) + \delta,$$

$$\delta = \frac{\epsilon}{\exp(3|x|)}, \quad \epsilon \sim N(0, 0.1^2)$$



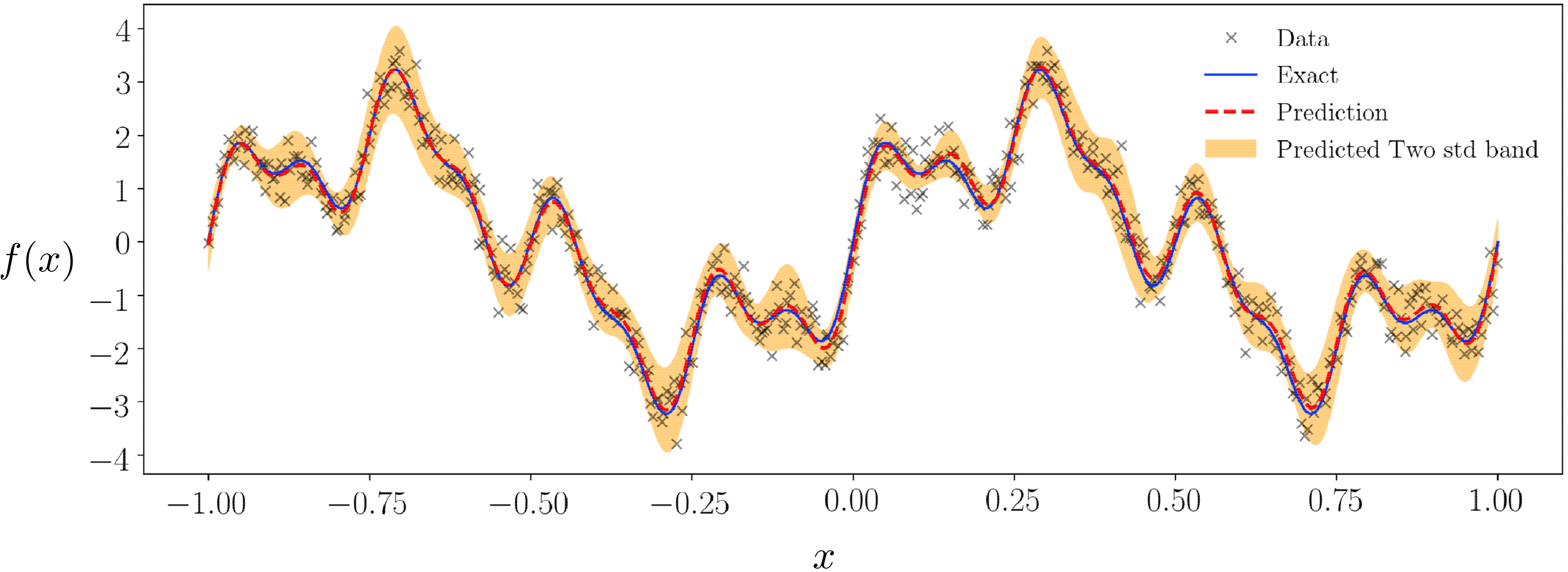




Neural nets: Feed-forward with 4 hidden layers, 50 neurons, tanh() activation, Adam optimizer.

Quantification of posterior uncertainty in deep learning

Synthetic noisy data: $y = 2\sin(2\pi x) + 8\sin(\pi x) + 0.5\sin(16\pi x) + \epsilon, \quad x \in [-1, 1], \quad \epsilon \sim \mathcal{N}(0, 0.3)$



Probabilistic model setup:

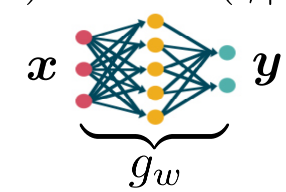
$$\boldsymbol{y} = g_w(\boldsymbol{x}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \gamma^{-1}),$$

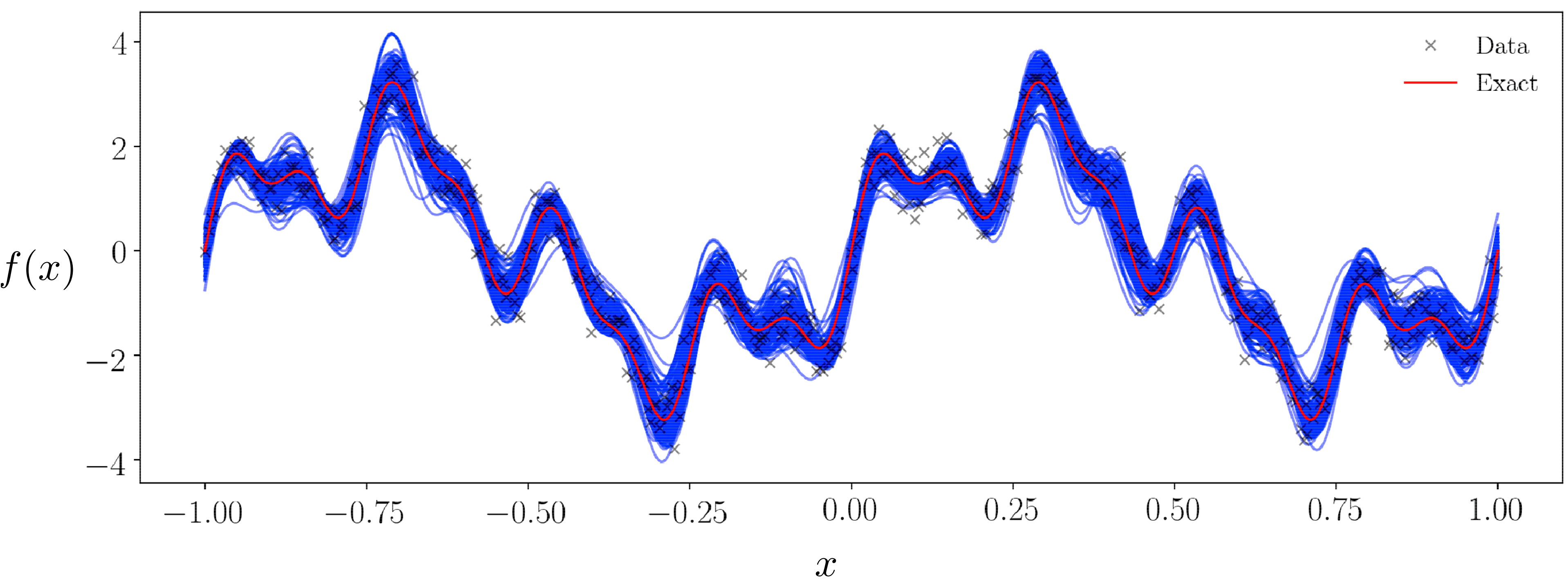
$$p(\boldsymbol{y}|\boldsymbol{x}, w, \gamma) = \prod_{i=1}^{N} \mathcal{N}(\boldsymbol{y}_i | g_w(\boldsymbol{x}_i), \gamma^{-1}),$$

$$p(w|\lambda) = \mathcal{N}(w|0, \lambda^{-1}),$$

$$p(\gamma) = \mathrm{Gam}(\gamma|5, 5),$$

$$p(\lambda) = \mathrm{Gam}(\gamma|5, 5),$$



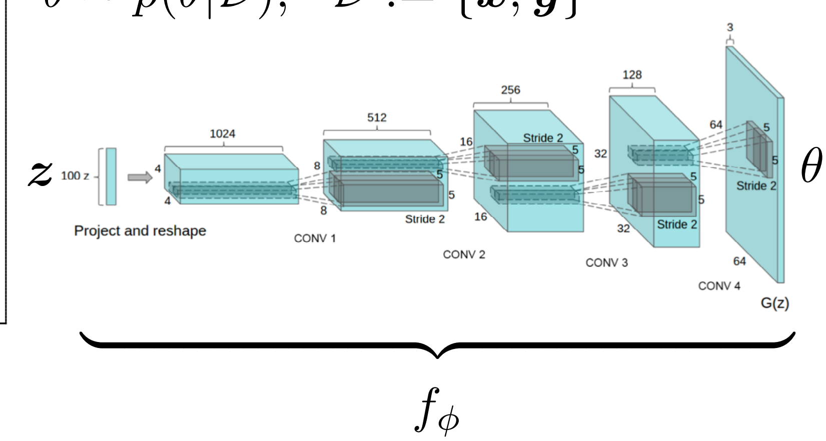


Posterior inference setup:

$$\theta := \{w, \lambda, \gamma\}$$

$$\theta = f_\phi(\boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z})$$

$$\theta \sim p(\theta|\mathcal{D}), \quad \mathcal{D} := \{\boldsymbol{x}, \boldsymbol{y}\}$$

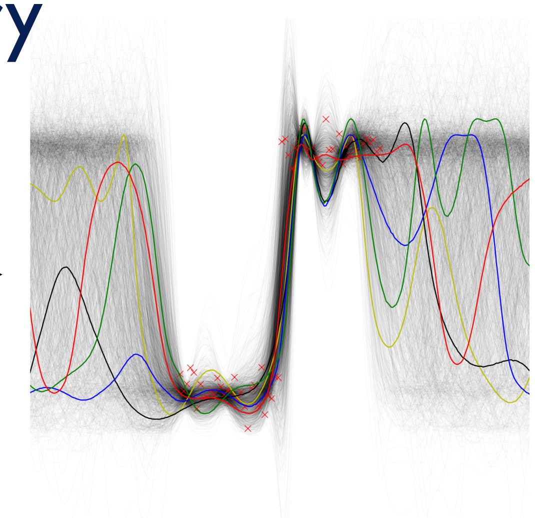


…or learning to sample the posterior of Bayesian neural networks, with neural networks!

Summary

$$z \sim p(z) \longrightarrow \boxed{u_t + \mathcal{N}_x u = 0} \longrightarrow p(u|x,t)$$

$$x, t \longrightarrow$$



Physics-informed deep generative models: $p(\boldsymbol{u}|\boldsymbol{x}, t, \boldsymbol{z}), \quad \boldsymbol{z} \sim p(\boldsymbol{z}), \quad \text{such that} \quad \boldsymbol{u}_t + \mathcal{N}_{\boldsymbol{x}}\boldsymbol{u} = 0.$

Advantages:

- Approximate arbitrarily complex and high-dimensional probability distributions.
- Bypasses the need for repeatedly sampling expensive experiments or numerical simulators.
- Encourage generative models to produce samples that satisfy PDEs.
- Avoid over-simplifying approximations (e.g. mean-field variational inference).
- Enables general and flexible schemes for statistical inference

Caveats:

- Adversarial models requires careful tuning.
- Theoretical asymptotic behavior is hard to be achieved in practice.

Yang, Y., & Perdikaris, P. (2019). Adversarial uncertainty quantification in physics-informed neural networks. Journal of Computational Physics.

Acknowledgements:

Yibo Yang (UPenn)

U.S. DEPARTMENT OF ENERGY

DARPA

Yang, Y., & Perdikaris, P. (2018). Physics-informed deep generative models. Neural Information Processing Systems, Workshop on Bayesian Deep Learning.

Yang, Y., & Perdikaris, P. (2019). Adversarial uncertainty quantification in physics-informed neural networks. Journal of Computational Physics.

Code: `https://github.com/PredictiveIntelligenceLab/UQPINNs`

Penn
UNIVERSITY of PENNSYLVANIA

Email: pgp@seas.upenn.edu