Physics-Informed Inference Time Scaling via Simulation-Calibrated Scientific Machine Learning

Yiping Lu

Northwestern | McCORMICK SCHOOL OF ENGINEERING

Joint work with Zexi Fan (PKU), Yan Sun (Gatech), Shihao Yang (Gatech)

Consider How you use ChatGPT…

What is the most efficient route from Times Square to Chelsea Market that minimizes walking?

Option 1: Subway

1. **Walk to a Nearby Subway Entrance:** Head to the Times Sq–42nd Street station.
2. **Board the E Train:** Catch an E train heading downtown (toward World Trade Center).
3. **Ride to 8th Avenue–14th Street Station:** Get off at this station (roughly 4 stops).
4. **Walk to Chelsea Market:** Exit the station and walk east on 14th Street for a few blocks until you reach Chelsea Market at 75 9th Avenue.

double-check with a map

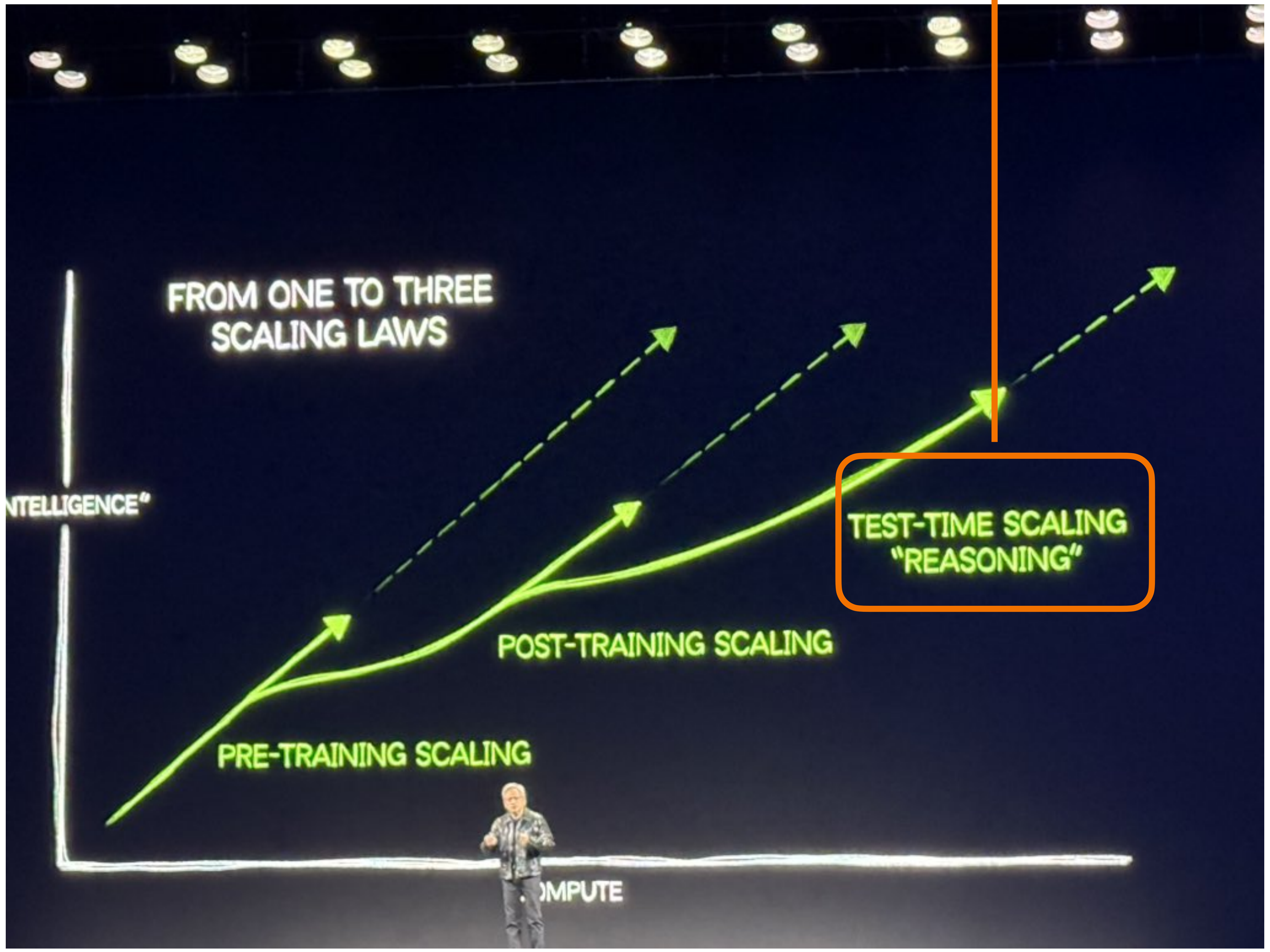
1. **At Times Square:** Enter the Times Square–42nd Street station.
2. **Board the 1 Train:** Hop on a downtown 1 train (the red line).
3. **Ride to 14th Street:** Stay on until you reach the 14th Street station.
4. **Exit Appropriately:** Use the exit that leads toward 9th Avenue—this drop-off point is just a short walk from Chelsea Market (located at 75 9th Ave).

Port Authortiy

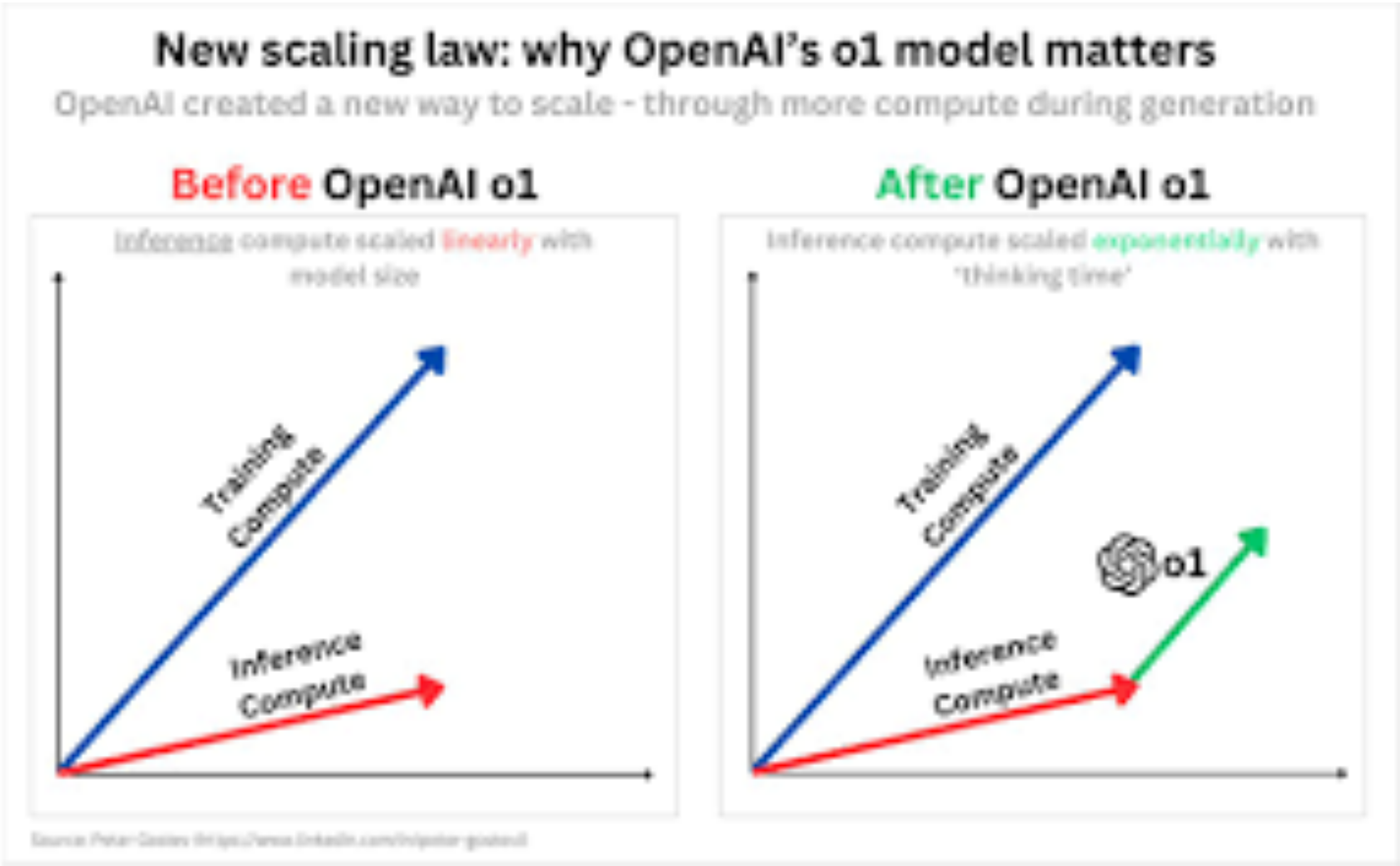
2 stops for A
3 stops for C/E

Inference Time Scaling Law

“No training”
e.g. answer question 10 times



Jensen Huang @CES 2025



How can we perform Inference-Time Scaling for Scientific Machine Learning?

With trustworthy garuntee

don’t fine-tune/retrain/add a new surrogate model

How can we perform Inference-Time Scaling for Scientific Machine Learning?

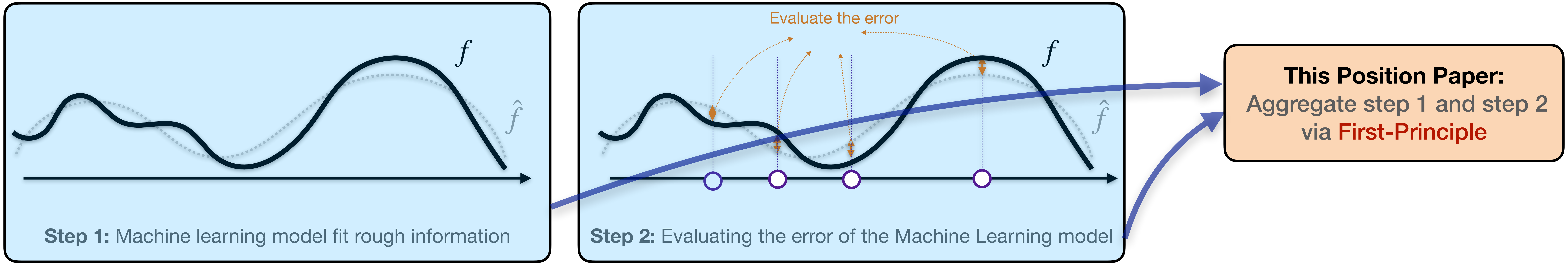
“Physics-informed”

With trustworthy garuntee

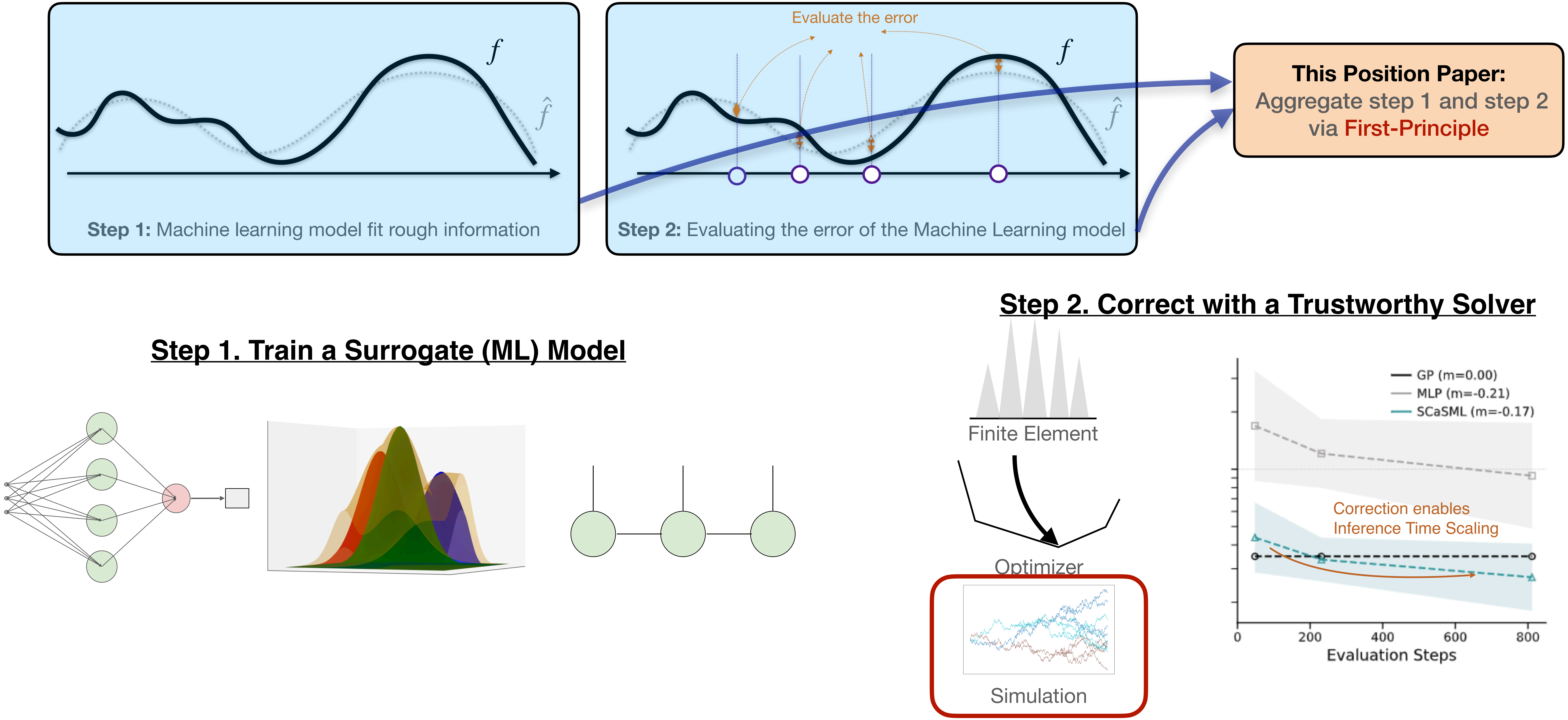
Idea: Debiasing using Feedback Information!

Hybrid Scientific Computing and Machine Learning

Physics-Informed Inference Time Scaling



Physics-Informed Inference Time Scaling



The Toy Example

Let's consider $\Delta u = f$

$$\{X_1, \cdots, X_n\} \sim \mathbb{P}_\theta \rightarrow \hat{\theta} \rightarrow \Phi(\hat{\theta})$$

Scientific Machine Learning

Downstream application

$$\theta = u, \quad \underbrace{X_i} = (x_i, f(x_i))$$

$$\Phi(\theta) = u(x), \text{ or } \int (u(x))dx$$

FEM/PINN/DGM/Tensor/Sparse Grid/…:

$\hat{\theta} = \hat{u}$

The Toy Example

Let's consider $\Delta u = f$

$$\{X_1, \cdots, X_n\} \sim \mathbb{P}_\theta \to \hat{\theta} \to \Phi(\hat{\theta})$$

Scientific Machine Learning

Downstream application

$$\theta = u, \quad \underbrace{X_i = }(x_i, f(x_i))$$

$\Phi(\theta) = u(x)$, or $\int (u(x))dx$

What is $\Phi(\theta) - \Phi(\hat{\theta}) = u(x) - \hat{u}(x)$?

FEM/PINN/DGM/Tensor/Sparse Grid/...:

$\hat{\theta} = \hat{u}$ $\longrightarrow$ $\Phi(\hat{\theta}) = \hat{u}(x)$

The Toy Example

Let's consider $\Delta u = f$

$$\{X_1, \cdots, X_n\} \sim \mathbb{P}_\theta \to \hat{\theta} \to \Phi(\hat{\theta})$$

Scientific Machine Learning

Downstream application

$\Delta u = f$

$\theta = u, \quad \underbrace{X_i} = (x_i, f(x_i))$

$\Phi(\theta) = u(x)$, or $\int (u(x))dx$

What is $\Phi(\theta) - \Phi(\hat{\theta}) = u(x) - \hat{u}(x)$?

$-$

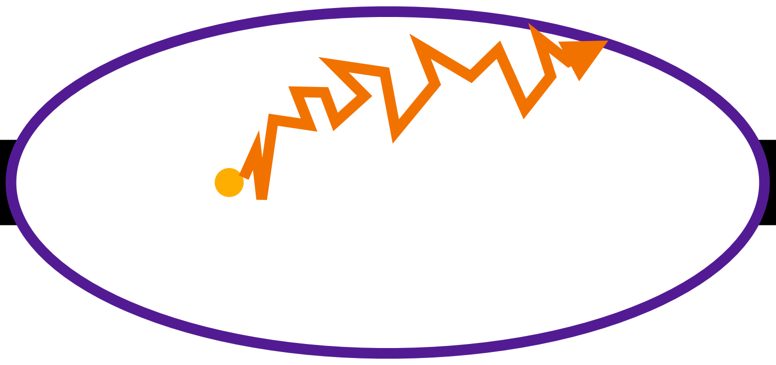
$\Delta \hat{u} = \hat{f}$

FEM/PINN/DGM/Tensor/Sparse Grid/...:

$\hat{\theta} = \hat{u} \longrightarrow \Phi(\hat{\theta}) = \hat{u}(x)$

$||$

$\Delta(u - \hat{u}) = f - \hat{f}$



$$(u - \hat{u})(x) = \mathbb{E} \int (f - \hat{f})(X_t)dt$$

Works for Semi-linear PDE

$$\frac{\partial U}{\partial t}(x,t) + \Delta U(x,t) + f(U(x,t)) = 0$$

Keeps the structure to enable brownian motion simulation

Can you do simulation for nonlinear equation?

Δ is linear!

Works for Semi-linear PDE

$$\frac{\partial U}{\partial t}(x,t) + \Delta U(x,t) + f(U(x,t)) = 0$$

Keeps the structure to enable brownian motion simulation

NN

$$\frac{\partial \hat{U}}{\partial t}(x,t) + \Delta \hat{U}(x,t) + f(\hat{U}(x,t)) = g(x,t)$$

$g(x,t)$ is the error made by NN

Works for Semi-linear PDE

$$\frac{\partial U}{\partial t}(x,t) + \Delta U(x,t) + f(U(x,t)) = 0$$

Keeps the structure to enable brownian motion simulation

NN

$$\frac{\partial \hat{U}}{\partial t}(x,t) + \Delta \hat{U}(x,t) + f(\hat{U}(x,t)) = g(x,t)$$

$g(x,t)$ is the error made by NN

Subtract two equations

Keeps the linear structure

Closed with respect to $U - \hat{U}$ for we know $\hat{U}$

$$\frac{\partial (U-\hat{U})}{\partial t}(x,t) + \Delta (U-\hat{U})(x,t)) + \underbrace{f(t, \hat{U}(x,t) + U(x,t) - \hat{U}(x,t)) - f(t, \hat{U}(x,t))}_{G\left(t, (U-\hat{U})(x,t)\right)} = g(x,t)\,.$$

How to simulate a Semi-linear PDE

MultiLevel Picard Iteration

$$\frac{\partial U}{\partial t}(x, t) + \Delta U(x, t) + f(U(x, t)) = 0$$

Keeps the structure to enable brownian motion simulation

Feyman-Kac $\longrightarrow$

$$U(x, t) := \mathbb{E}\left[\int_s^T f(U(W_t, t))dt\right]$$

Brownian Motion

hard to simulate for we don't know U

How to simulate a Semi-linear PDE

MultiLevel Picard Iteration

$$\frac{\partial U}{\partial t}(x,t) + \Delta U(x,t) + f(U(x,t)) = 0$$

Keeps the structure to enable brownian motion simulation

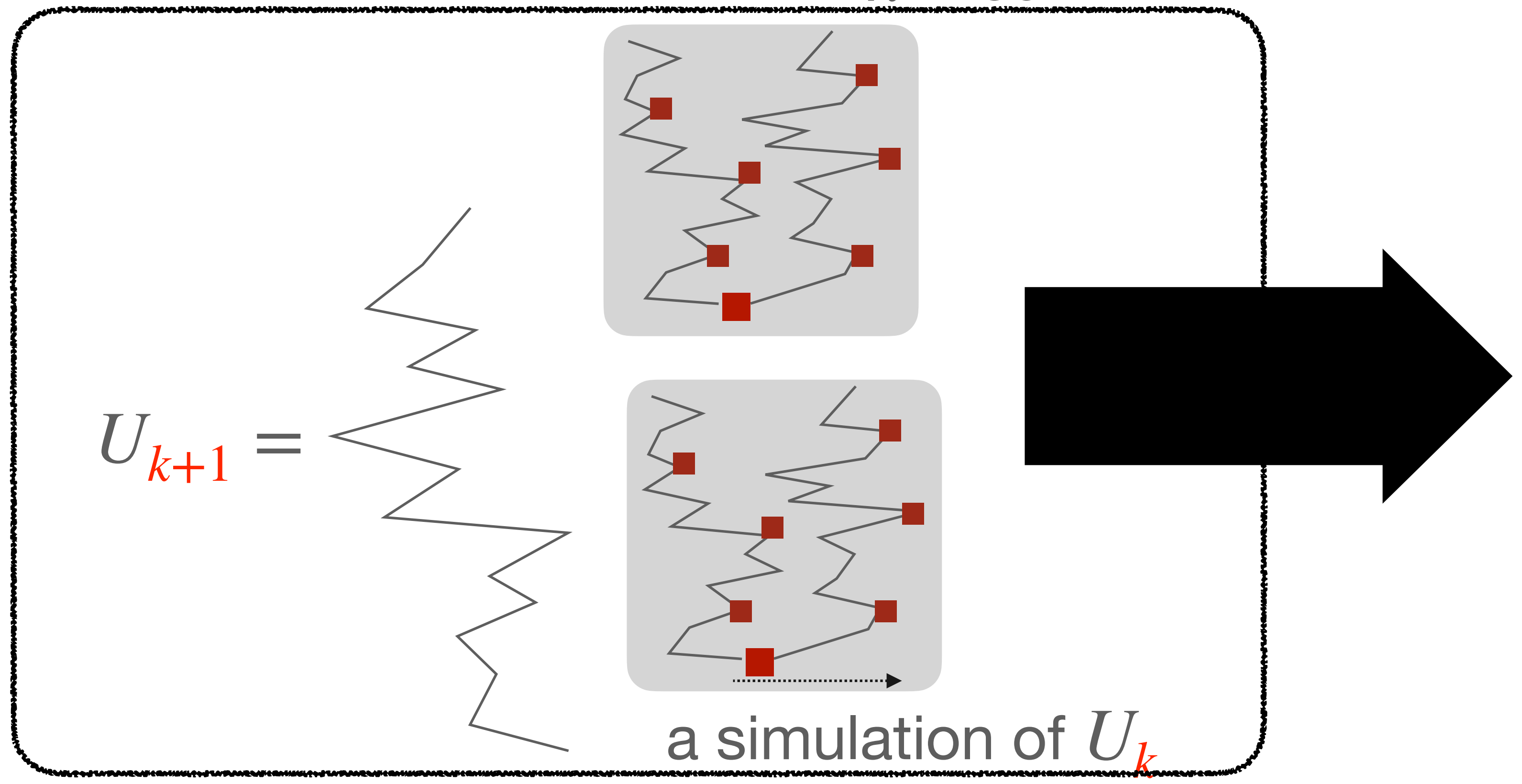
Feyman-Kac

Brownian Motion

$$U_{k+1}(x,t) := \mathbb{E}\left[\int_s^T f(U_k(W_t, t))dt\right]$$

Idea: Using Picard Iteration turn to a Nested Simulation Problem

$$\lim_{k\to\infty} U_k = U$$

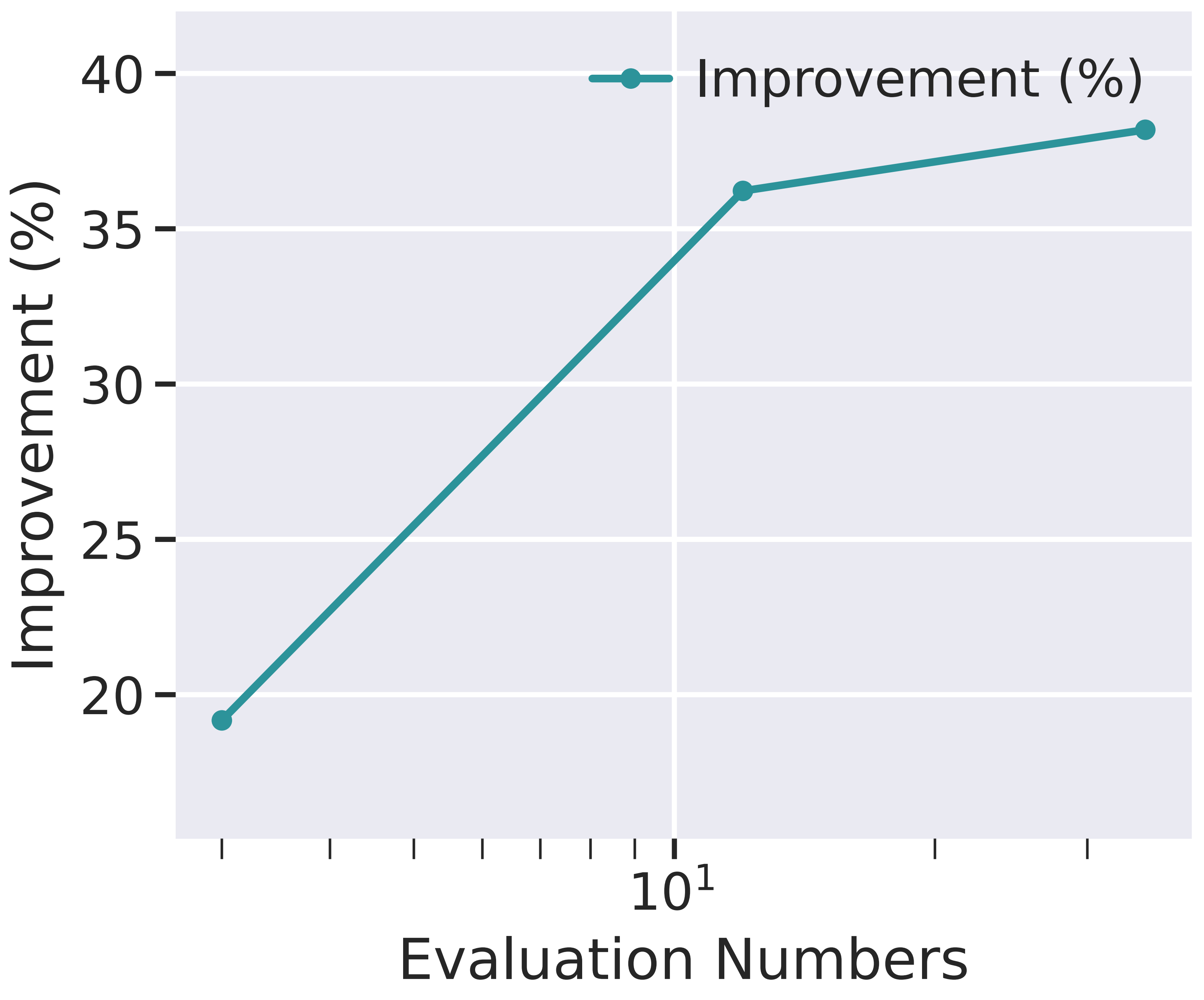


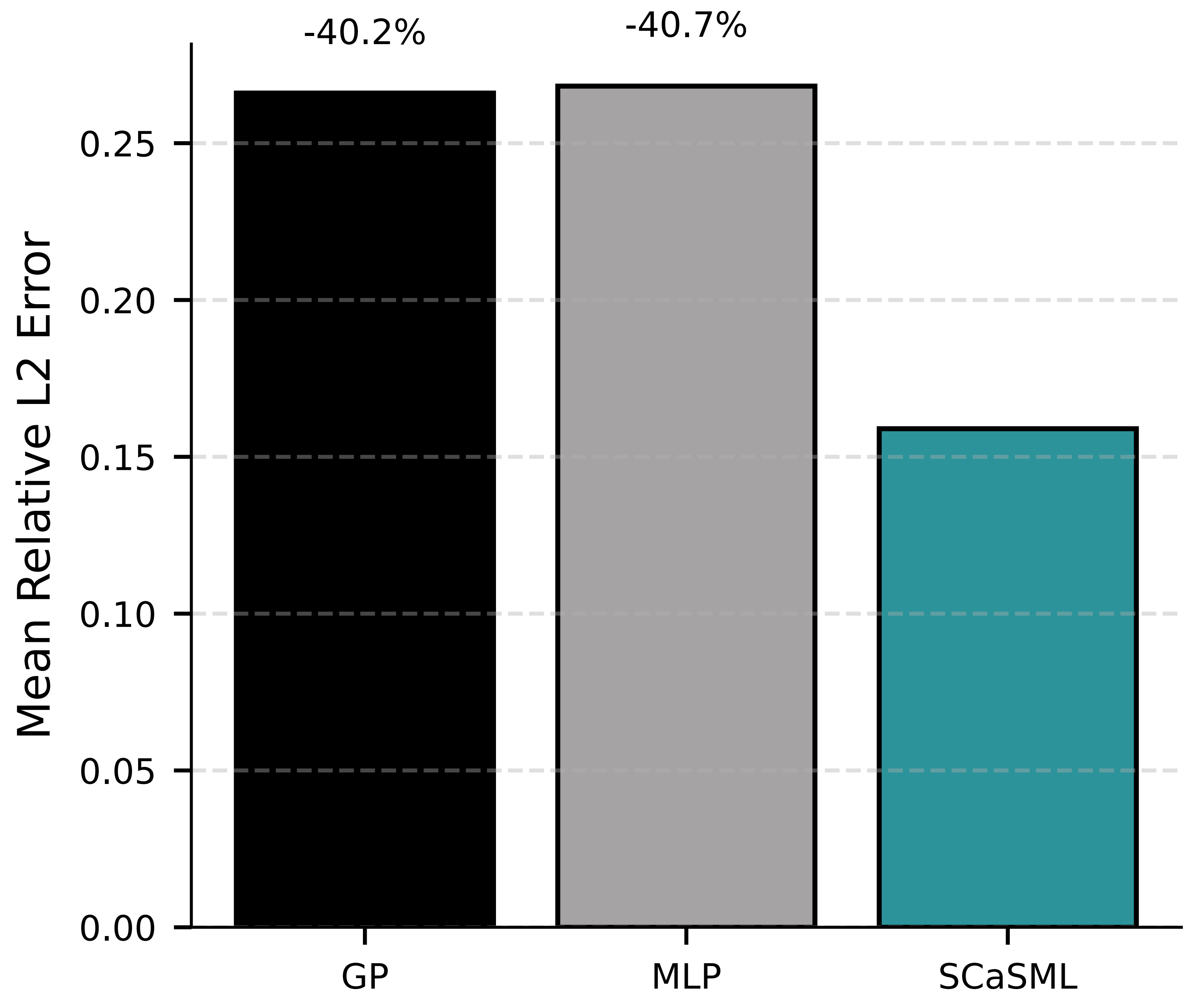
Multileve Monte Carlo

Inference-Time Scaling

$$\frac{\partial}{\partial t}u + \left[\sigma^2 u - \frac{1}{d} - \frac{\bar{\sigma}^2}{2}\right](\nabla \cdot u) + \frac{\bar{\sigma}^2}{2}\Delta u = 0$$

have closed-form solution $g(x) = \frac{\exp(T + \sum_i x_i}{1 + \exp(T + \sum_i x_i)}$





Method	Convergence Rate
PINN	$O(n^{-s/d})$
MLP	$O(n^{-1/4})$
ScaSML	$O(n^{-1/4-s/d})$

Why SCaSML can leads to **Improved** Rate

$\Delta u = f$ - $\Delta \hat{u} = \hat{f}$

$X_i = (x_i, f(x_i))$ → **FEM/PINN/ Tensor/Sparse Grid/...:** $\hat{\theta} = \hat{u}$

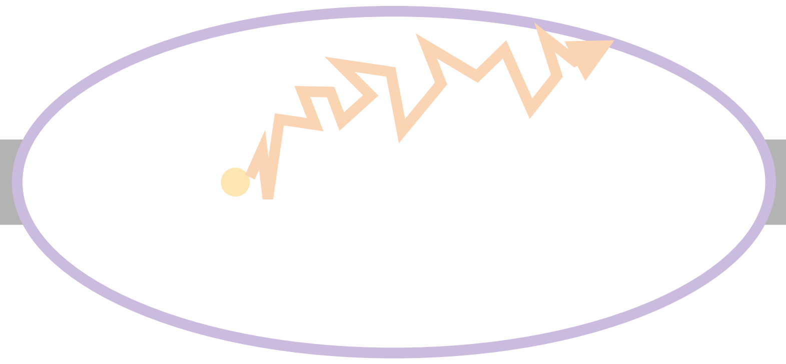
Assume a convergence rate in phase 1

using n collocation points:

$$\|f - \hat{f}\| = O(n^{-\alpha})$$

What is $\Phi(\theta) - \Phi(\hat{\theta}) = u(x) - \hat{u}(x)$?

$\Delta(u - \hat{u}) = f - \hat{f}$ →



→

$$(u - \hat{u})(x) = \mathbb{E}\int \underbrace{(f - \hat{f})}_{}(X_t)dt$$

Variance is $\|f - \hat{f}\|^2$

using NN as a *Control Variate*!

Final Simulation Error

using n simulations:

$$\sqrt{\frac{\text{Variance}}{n}} = O\left(\frac{n^{-2\alpha}}{n}\right) = n^{-1/2-\alpha}$$

Why SCaSML can leads to Improved Rate

$\Delta u = f$ − $\Delta \hat{u} = \hat{f}$

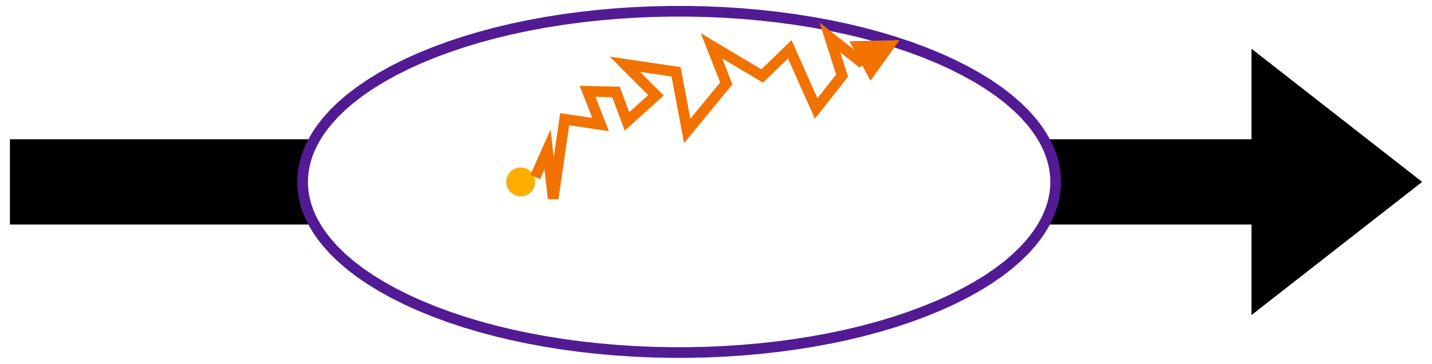
$X_i = (x_i, f(x_i))$ ⟶ **FEM/PINN/ Tensor/Sparse Grid/…:** $\hat{\theta} = \hat{u}$

Assume a convergence rate in phase 1 using n collocation points:

$$\|f - \hat{f}\| = O(n^{-\alpha})$$

What is $\Phi(\theta) - \Phi(\hat{\theta}) = u(x) - \hat{u}(x)$?

$$\Delta(u - \hat{u}) = f - \hat{f}$$



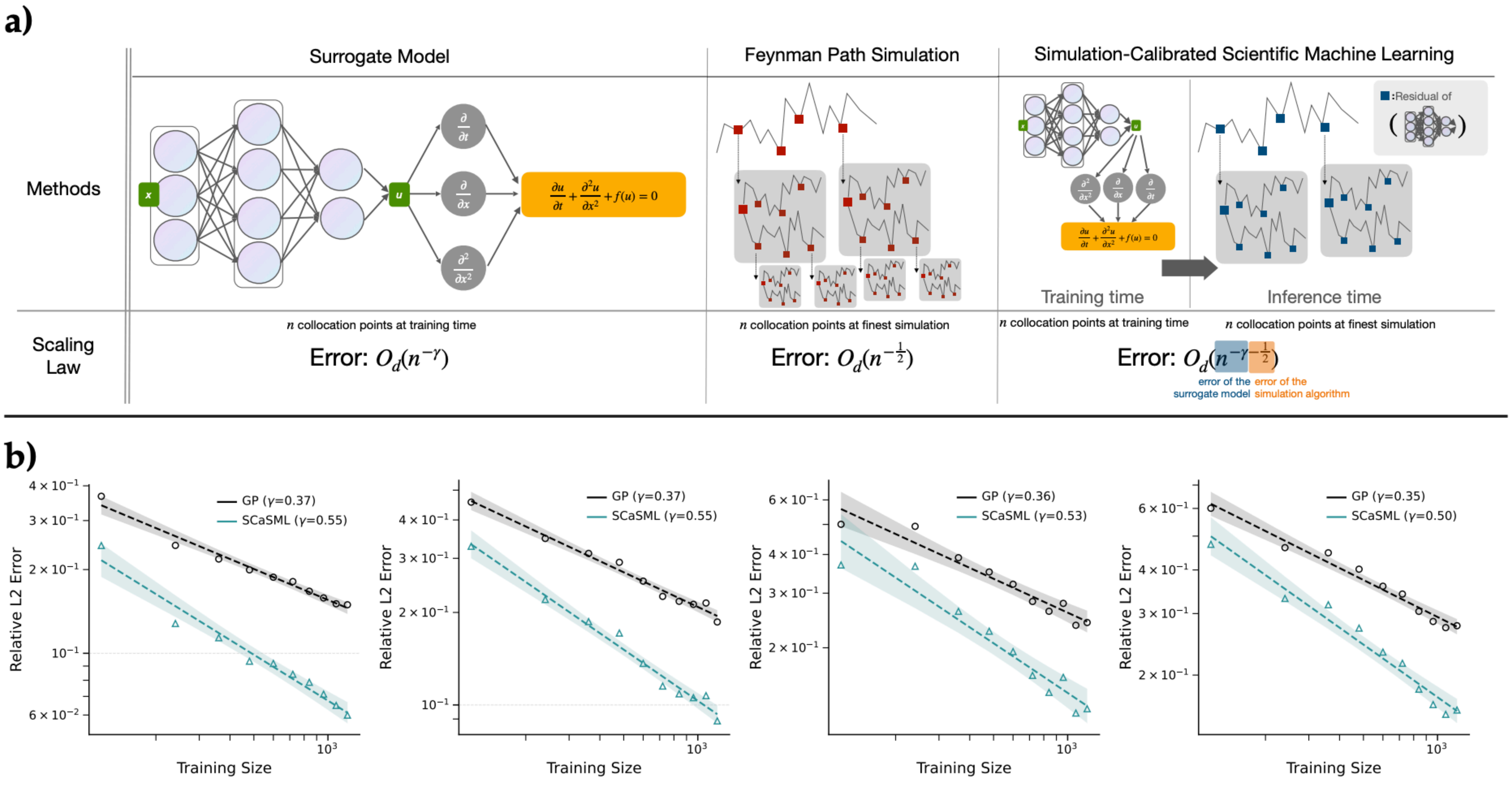
$$(u - \hat{u})(x) = \mathbb{E}\int \overbrace{(f - \hat{f})}^{\text{Variance is } \|f - \hat{f}\|^2}(X_t)dt$$

using NN as a *Control Variate*!

Final Simulation Error using n simulations:

$$\sqrt{\frac{\text{Variance}}{n}} = O\left(\frac{n^{-2\alpha}}{n}\right) = n^{-1/2-\alpha}$$

Better Scaling Law



(a) $d = 20$ (b) $d = 40$ (c) $d = 60$ (d) $d = 80$

Numerical Results

		Time (s)			Relative L^2 Error			L^∞ Error			L^1 Error		
		SR	MLP	SCaSML	SR	MLP	SCaSML	SR	MLP	SCaSML	SR	MLP	SCaSML
LCD	10d	2.64	11.24	23.75	5.24E-02	2.27E-01	**2.73E-02**	2.50E-01	9.06E-01	**1.61E-01**	3.43E-02	1.67E-01	**1.78E-02**
	20d	1.14	7.35	17.59	9.09E-02	2.35E-01	**4.73E-02**	4.52E-01	1.35E+00	**3.28E-01**	9.47E-02	2.37E-01	**4.52E-02**
	30d	1.39	7.52	25.33	2.30E-01	2.38E-01	**1.84E-01**	4.73E+00	1.59E+00	**1.49E+00**	**1.75E-01**	2.84E-01	1.91E-01
	60d	1.13	7.76	35.58	3.07E-01	2.39E-01	**1.32E-01**	3.23E+00	2.05E+00	**1.55E+00**	5.24E-01	4.07E-01	**2.06E-01**
VB-PINN	20d	1.15	7.05	13.82	1.17E-02	8.36E-02	**3.97E-03**	3.16E-02	2.96E-01	**2.16E-02**	5.37E-03	3.39E-02	**1.29E-03**
	40d	1.18	7.49	16.48	3.99E-02	1.04E-01	**2.85E-02**	8.16E-02	3.57E-01	**7.16E-02**	1.97E-02	4.36E-02	**1.21E-02**
	60d	1.19	7.57	19.83	3.97E-02	1.17E-01	**2.90E-02**	8.10E-02	3.93E-01	**7.10E-02**	1.95E-02	4.82E-02	**1.24E-02**
	80d	1.32	7.48	21.99	6.78E-02	1.19E-01	**5.68E-02**	1.89E-01	3.35E-01	**1.79E-01**	3.24E-02	4.73E-02	**2.49E-02**
VB-GP	20d	1.97	10.66	65.46	1.47E-01	8.32E-02	**5.52E-02**	3.54E-01	**2.22E-01**	2.54E-01	7.01E-02	3.50E-02	**1.91E-02**
	40d	1.68	10.14	49.38	1.81E-01	1.05E-01	**7.95E-02**	4.01E-01	3.47E-01	**3.01E-01**	9.19E-02	4.25E-02	**3.43E-02**
	60d	1.01	7.25	35.14	2.40E-01	2.57E-01	**1.28E-01**	3.84E-01	9.50E-01	**7.10E-02**	1.27E-01	9.99E-02	**6.11E-02**
	80d	1.00	7.00	38.26	2.66E-01	3.02E-01	**1.52E-01**	3.62E-01	1.91E+00	**2.62E-01**	1.45E-01	1.09E-01	**7.59E-02**
LQG	100d	1.54	8.67	26.95	7.96E-02	5.63E+00	**5.51E-02**	7.78E-01	1.26E+01	**6.78E-01**	1.40E-01	1.21E+01	**8.68E-02**
	120d	1.25	8.17	27.46	9.37E-02	5.50E+00	**6.64E-02**	9.02E-01	1.27E+01	**8.02E-01**	1.73E-01	1.22E+01	**1.05E-01**
	140d	1.80	8.27	29.72	9.79E-02	5.37E+00	**6.78E-02**	1.00E+00	1.27E+01	**9.00E-01**	1.91E-01	1.23E+01	**1.11E-01**
	160d	1.74	9.07	32.08	1.11E-01	5.27E+00	**9.92E-02**	1.38E+00	1.28E+01	**1.28E+00**	2.15E-01	1.23E+01	**1.79E-01**
DR	100d	1.62	7.75	60.86	9.52E-03	8.99E-02	**8.87E-03**	7.51E-02	6.37E-01	**6.51E-02**	1.13E-02	9.74E-02	**1.11E-02**
	120d	1.26	7.28	65.66	1.11E-02	9.13E-02	**9.90E-03**	7.10E-02	5.74E-01	**6.10E-02**	1.40E-02	9.97E-02	**1.23E-02**
	140d	2.38	7.82	76.90	3.17E-02	8.97E-02	**2.94E-02**	1.79E-01	8.56E-01	**1.69E-01**	3.96E-02	9.77E-02	**3.67E-02**
	160d	1.75	7.42	82.40	3.46E-02	9.00E-02	**3.23E-02**	2.08E-01	8.02E-01	**1.98E-01**	4.32E-02	9.75E-02	**4.02E-02**

SCaSML
Simulation-Calibrated
Scientific Machine Learning

Arxiv

Physics-Informed Inference Time Scaling via Simulation-Calibrated Scientific Machine Learning

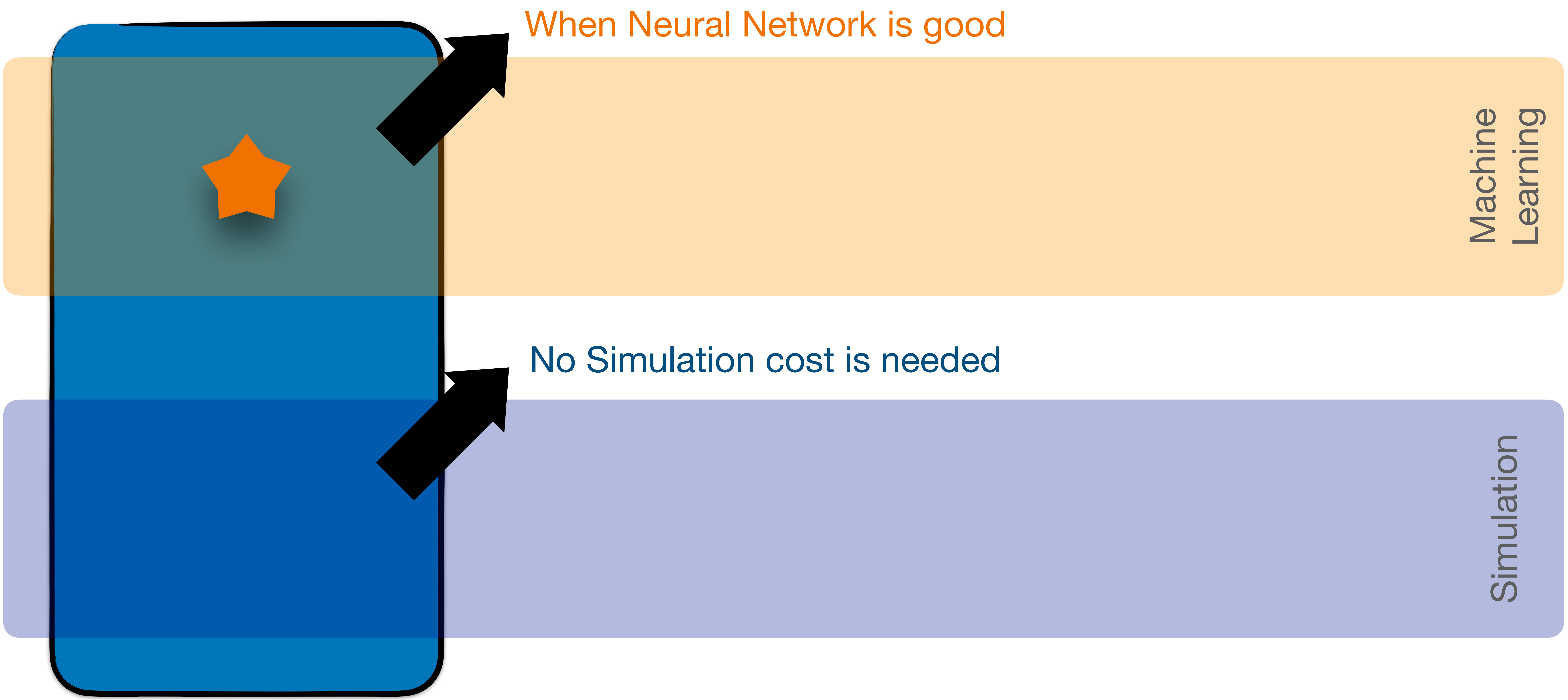
Zexi Fan[1], Yan Sun [2], Shihao Yang[3], Yiping Lu*[4]

[1] Peking University [2] Visa Inc. [3] Georgia Institute of Technology [4] Northwestern University

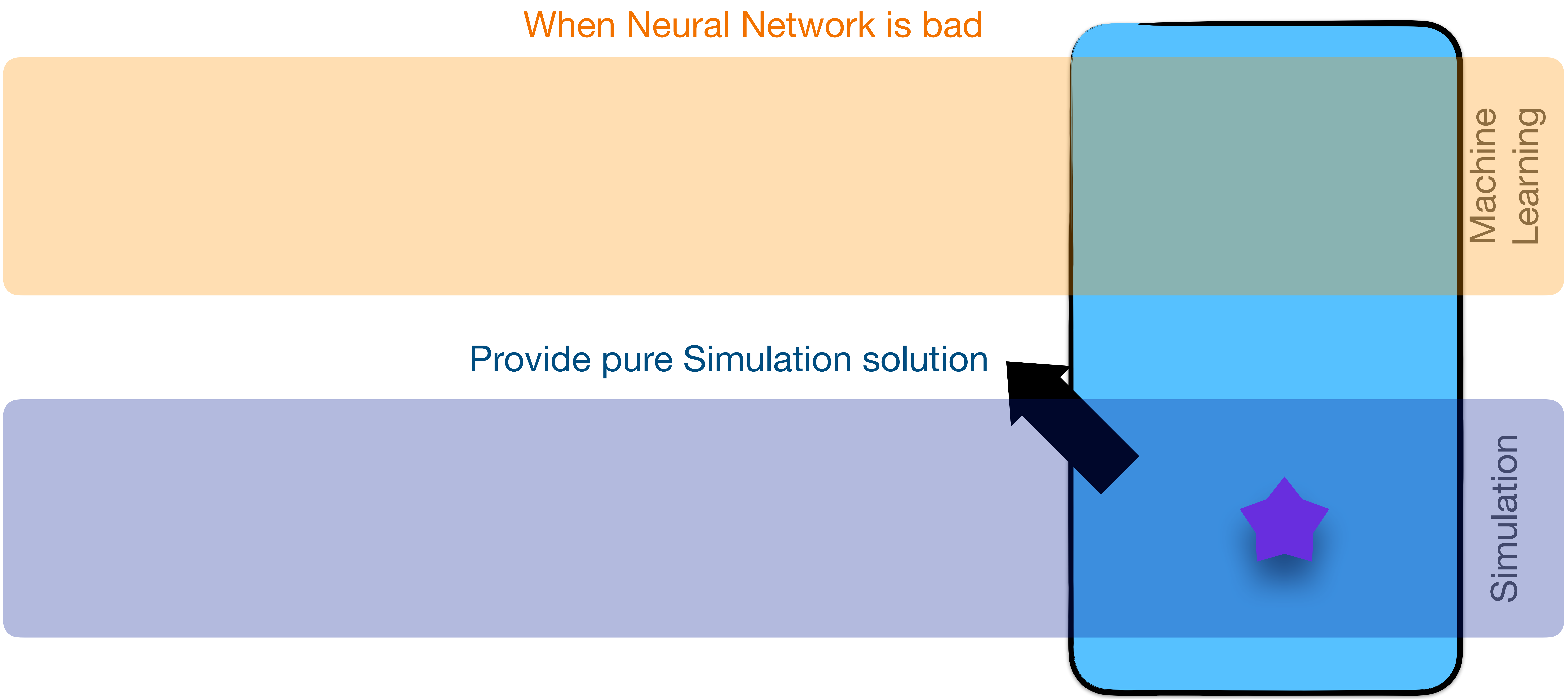
fanzexi_francis@stu.pku.edu.cn,yansun414@gmail.com,
shihao.yang@isye.gatech.edu,yiping.lu@northwestern.edu

https://2prime.github.io/files/scasml_techreport.pdf

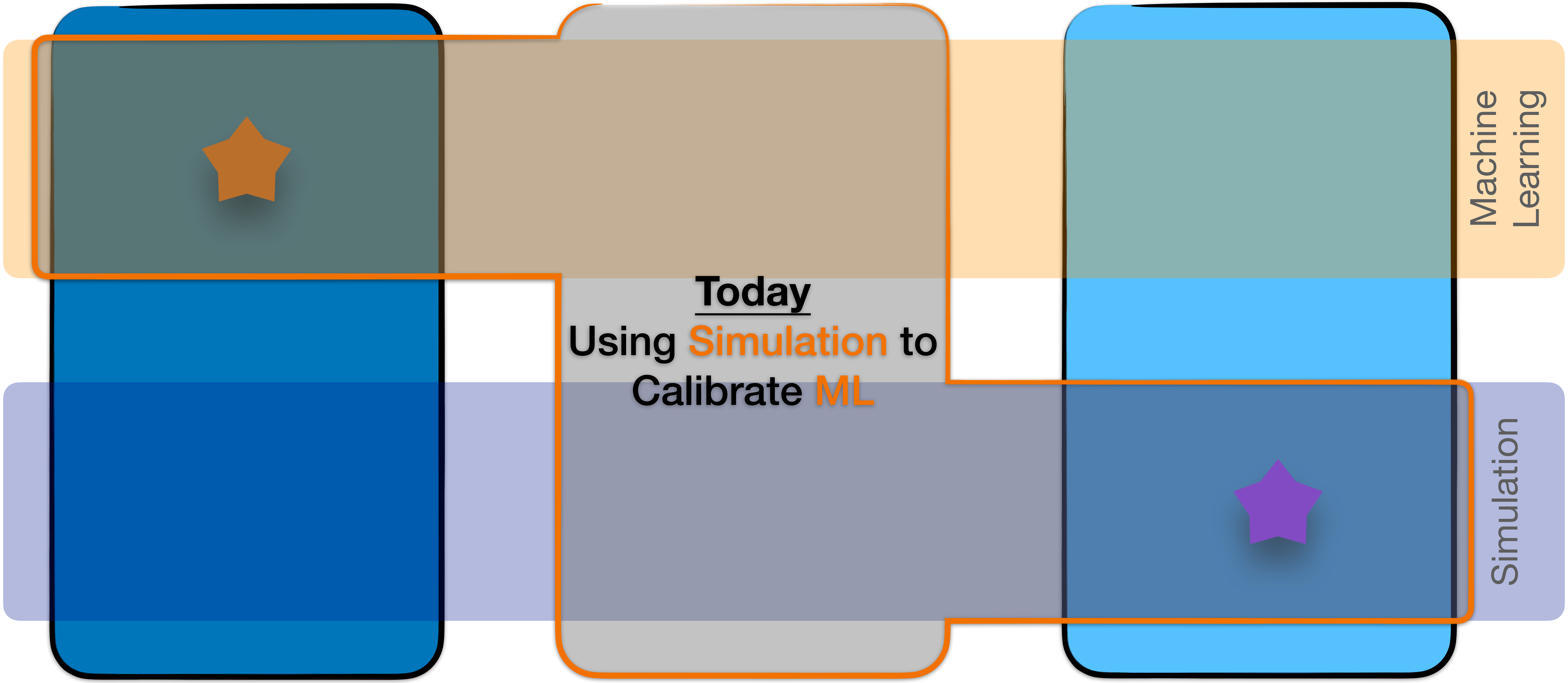
Our Aim Today : A Marriage



Our Aim Today : A Marriage



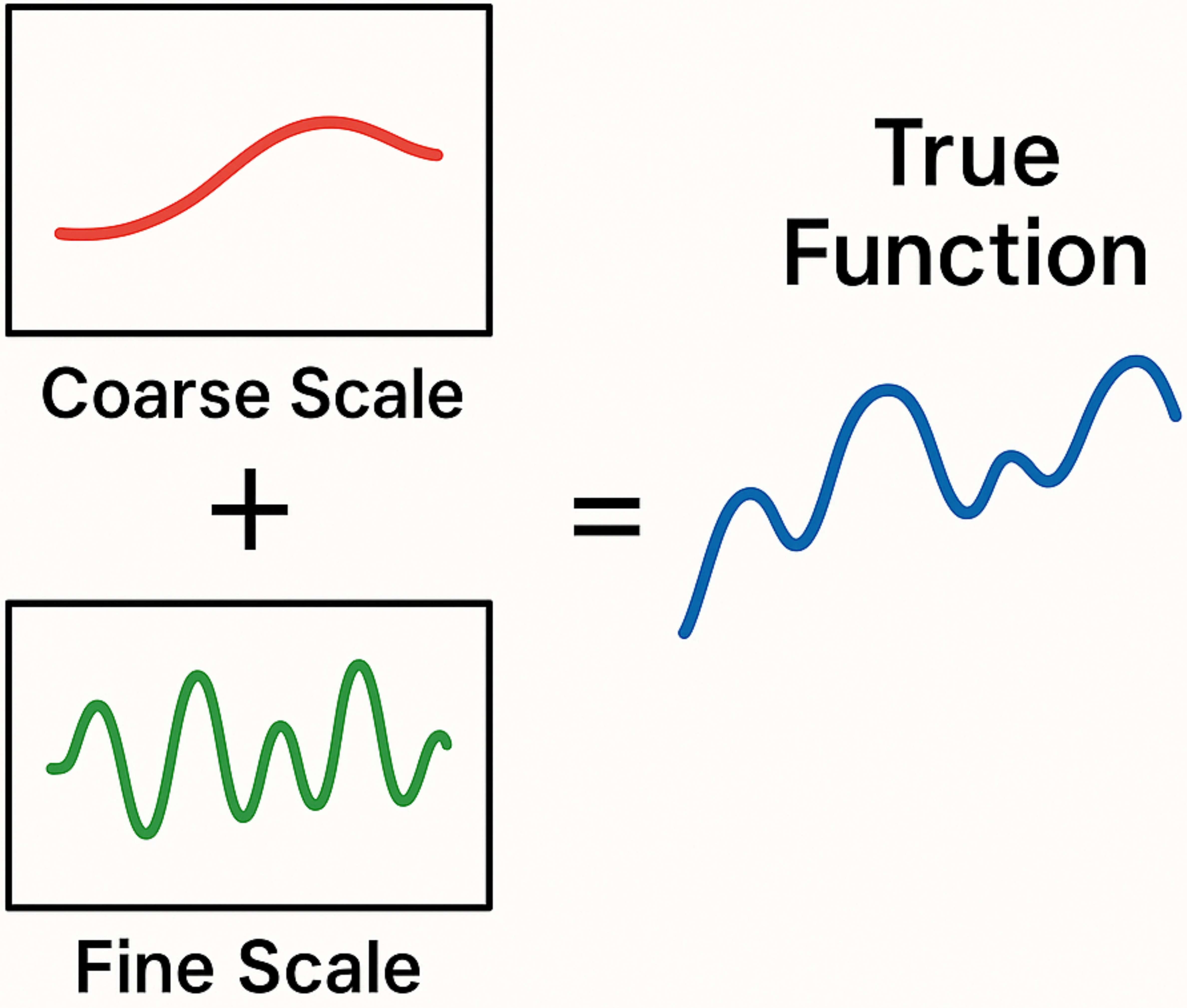
Our AIM Today: A Marriage



A multiscale view

Capture via surrogate model

Capture via Monte-Carlo



More Examples…

$$\{X_1, \cdots, X_n\} \sim \mathbb{P}_\theta \rightarrow \hat{\theta} \rightarrow \Phi(\hat{\theta})$$

Scientific Machine Learning — Downstream application

Example 1 $\theta = f, \quad X_i = (x_i, f(x_i))$ $\quad \Phi(\theta) = \int f^q(x)dx$

Blanchet J, Chen H, Lu Y, et al. When can regression-adjusted control variate help? rare events, sobolev embedding and minimax optimality. Advances in Neural Information Processing Systems, 2023, 36: 36566-36578.

Provides minmax optimality

Example 2 $\theta = \Delta^{-1} f, \quad X_i = (x_i, f(x_i))$ $\quad \Phi(\theta) = \theta(x)$

Example 3 $\theta = A, \quad X_i = (x_i, Ax_i)$ $\quad \Phi(\theta) = \mathrm{tr}(A)$

Estimation $\hat{A}$ via Randomized SVD — Estimate $\mathrm{tr}(A - \hat{A})$ via Hutchinson's estimator

More Examples… (Uncertainty Quantification)

$$\{X_1, \cdots, X_n\} \sim \mathbb{P}_\theta \rightarrow \hat{\theta} \rightarrow \Phi(\hat{\theta})$$

Scientific Machine Learning

Downstream application

Example 5

$$\theta = \theta, \quad X_i \sim P_\theta$$

Confiednece Interval of Point Estimation

Quantile regression

Conformal Prediction

Romano Y, Patterson E, Candes E. Conformalized quantile regression. Neurips 2019.

Influnce Function

Bootstrap

Liu K, Blanchet J, Ying L, et al. Orthogonal bootstrap: efficient simulation of input uncertainty. ICML 2024.

LLM

Taylor Expansion

Angelopoulos A N, Bates S, Fannjiang C, et al. Prediction-powered inference. Science, 2023

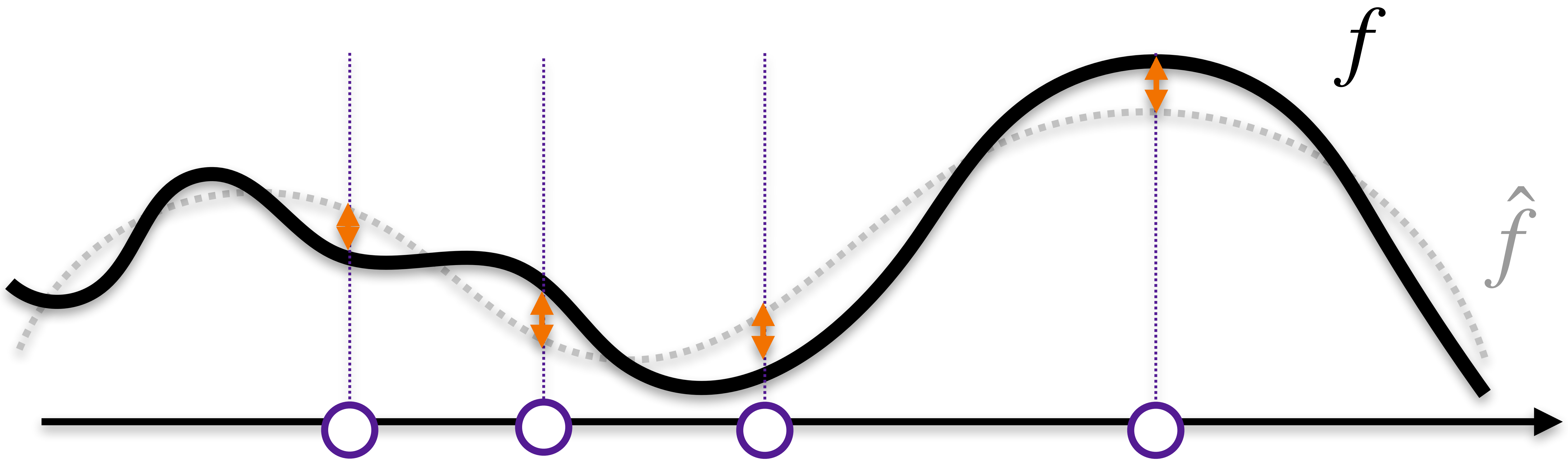
What is SCaSML about?

SCaSML

$$\{X_1, \cdots, X_n\} \sim \mathbb{P}_\theta \rightarrow \theta \rightarrow \Phi(\theta)$$

Step 1: Using Machine Learning to fit the rough function/environment

Step 2: Using validation dataset to know how much mistake machine learning algorithm has made



Step 3: Using Simulation algorithm to estimate $\Phi(\theta) - \Phi(\hat{\theta})$

Using ML surrogate during inference time to improve ML solution